



UNIVERSIDADE NOVA LISBOA

Faculdade Ciências Tecnologia

Daniel Cruz Veiga Álvares Pereira

LICENCIADO EM CIÊNCIAS DA ENGENHARIA

Elementos finitos híbridos com refinamento p -adaptativo para modelação de problemas de transferência de calor

Dissertação para a obtenção do Grau de Mestre em
Engenharia Civil – Perfil de Estruturas

Orientador: Prof^ª. Doutora Ildi Cismaşiu
Co-Orientador: Prof. Doutor Dragoş Ionuţ Moldovan

Júri:

Presidente: Prof. Doutor Nuno da Costa Guerra
Arguente: Prof. Doutor José Paulo Baptista Moitinho Almeida
Vogal: Prof^ª. Doutora Ildi Cismaşiu

MAIO, 2015



FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

“Copyright” Daniel Cruz Veiga Álvares Pereira, FCT/UNL e UNL

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Resumo

Neste trabalho é abordada a modelação da transferência de calor por condução, em duas dimensões e regime permanente, utilizando métodos computacionais de alto desempenho, aliados a algoritmos que reduzem significativamente a complexidade da utilização dos mesmos. O método empregue é baseado no modelo de temperatura da formulação híbrida do método de elementos finitos, para o qual é desenvolvido um algoritmo de refinamento p -adaptativo. Uma interface gráfica é implementada para facilitar a introdução de dados. Na formulação híbrida do método dos elementos finitos, são aproximados, de forma independente, os campos de temperaturas nos domínios dos elementos e os fluxos de calor normais nas fronteiras de Dirichlet e interiores. As aproximações são construídas hierarquicamente, utilizando bases polinomiais, e não dependem dos nós da malha de elementos finitos. As funções de forma são utilizadas para impor, em forma fraca, a equação de Fourier nos domínios dos elementos e as equações de continuidade da temperatura nas fronteiras essenciais. A formulação oferece flexibilidade considerável na escolha dos graus das funções de aproximação, tanto no domínio como nas fronteiras, sendo trivial (ao contrário dos elementos conformes de temperatura) combinar elementos com vários refinamentos de base na mesma malha. Ademais, o refinamento das bases de aproximação (refinamento p) não obriga a recalcular todas as funções de forma, tornando assim possível reutilizar os coeficientes do sistema resolutivo associados ao refinamento anterior. Esta propriedade, aliada às elevadas ordens de convergência obtidas através do refinamento p , torna interessante a implementação de técnicas automáticas de refinamento p -adaptativo, cujas implementação e avaliação formam o cerne deste trabalho. A técnica de refinamento p -adaptativo que se apresente permite seleccionar, de forma criteriosa e coerente, regiões onde as bases de aproximação são adequadamente enriquecidas, explorando a natureza hierárquica das mesmas. Três critérios para a escolha dessas regiões são propostos, e os seus desempenhos avaliados e comparados utilizando três situações práticas de características e níveis de exigência numérica bastante diferentes. Apesar de haver diferenças consideráveis entre os seus desempenhos numéricos, todos os critérios de refinamento levam a uma escolha automatizada e convergente dos graus das bases de aproximação do campo de temperaturas no domínio e dos fluxos nas fronteiras essenciais. Neste caso, o critério de convergência do algoritmo baseia-se na variação de energia térmica total da malha de elementos finitos, em iterações sucessivas.

Palavras Chave:

Transferência de calor;

Condução de calor;

Elementos Finitos Híbridos;

Refinamento p -adaptativo.

Abstract

This work is centred on the finite element modelling of steady-state heat conduction in two dimensions. The model is constructed using high-performance finite elements, supported by algorithms designed to simplify their use by inexperienced analysts. The temperature model of the hybrid finite element formulation is used to solve the governing boundary value problem and its implementation is enhanced using a fully automatic p -adaptive algorithm for the selection of the domain and boundary approximations. A graphical user interface is also implemented to support data input. Hybrid finite elements approximate independently two (dual) solutions fields, one in the domain of the finite element and the other on its essential boundary. The temperature model used in this work approximates the temperature field in the domain of each element and the normal flux field on the Dirichlet and interior boundaries of the mesh. All approximations are built hierarchically, using polynomial bases not linked in any way to the nodes of the mesh. The shape functions are used to enforce weakly the Fourier law in the domain of the elements and the temperature continuity on its essential boundaries. The formulation offers considerable flexibility to the choice of the degrees of the polynomial bases. As opposed to conforming temperature elements, in hybrid formulations it is straightforward to combine elements with different basis refinements in the same mesh. Moreover, incrementing the refinement of the approximation bases (p -refinement) does not require the recalculation of the shape functions, thus enabling the reuse of all solving system coefficients from the previous refinement. This property, supported by the high orders of convergence rate of the p -refinement process, justifies the investment into the implementation of the automated p -refinement technique that stands at the core of this work. The algorithm implemented here enables the automatic selection of the regions where the approximation functions should be enriched. The selection is based on three criteria, whose traits and performances are assessed and compared using three practical case studies of different structural characteristics and modelling difficulty. Despite considerable differences in their numerical behaviour, all criteria successfully lead to a convergent and automatic choice of the degrees of the domain and boundary bases. The convergence criterion is based on the assessment of the variation of the total thermal energy in successive iterations.

Keywords:

Heat Transfer;

Heat Conduction;

Hybrid Finite Elements;

p -Adaptive refinement.

Agradecimentos

Em primeiro lugar, desejo agradecer à minha orientadora, Professora Doutora Ildi Cismaşiu pela sempre presente disponibilidade e transmissão de conhecimentos no decurso deste trabalho. Em igual medida, quero também estender os meus imensuráveis agradecimentos ao meu co-orientador, Professor Doutor Ionuţ Dragos Moldovan pela imensa paciência e cuidado na exposição de material enquanto navegava em águas desconhecidas. Por fim, num tom mais pessoal, agradecer a todos os colegas, e amigos que, de uma forma ou de outra, me acompanharam durante o meu percurso académico, em todos os altos e baixos. E claro, à minha família.

Conteúdo

Resumo	iii
Abstract	v
Agradecimentos	vi
Lista de Figuras	xi
Lista de Tabelas	xiii
Lista de Abreviaturas e Símbolos	1
1 Introdução	1
1.1 Motivação	1
1.2 Objectivos do Trabalho	2
1.3 Estado da Arte	3
1.4 Organização do Trabalho	8
2 Fundamentos Teóricos	11
2.1 Mecanismos de Transferência	11
2.1.1 Condução	12
2.1.2 Convecção	16
2.1.3 Radiação	17
2.2 Condições de fronteira	18
2.3 Método dos Resíduos Pesados	19
2.4 Formulação em elementos híbridos	22
2.4.1 Equações do modelo de temperaturas da formulação híbrida	22
2.4.2 Indeterminação estática e cinemática	26
3 Implementação Computacional	29
3.1 Metodologia geral	29
3.2 Hipóteses e simplificações	30
3.3 Implementação	31

3.3.1	Geometria e geração de malha	31
3.3.2	Funções de forma	32
3.3.3	Montagem do Sistema	34
3.3.4	Campos de Soluções	36
3.3.5	Integração Numérica	37
3.4	Estrutura de Dados Topológica	38
4	Refinamento p-Adaptativo	41
4.1	Aspectos gerais	41
4.1.1	Origem do erro	42
4.1.2	Estratégia de refinamento	43
4.1.3	Refinamento na fronteira	44
4.1.4	Refinamento no domínio	47
4.1.5	Critérios de selecção de refinamento	50
4.2	Descrição do algoritmo	51
4.3	Aspectos de implementação	54
5	Resultados Numéricos	57
5.1	Caso de estudo 1 - problema bisimétrico	57
5.2	Caso de estudo 2 - problema assimétrico	65
5.3	Caso de estudo 3 - problema simétrico	75
6	Conclusões	85
6.1	Conclusões	85
6.2	Desenvolvimentos Futuros	86
	Bibliografia	91
	Apêndice A Equação governativa do problema na forma fraca	94
	Apêndice B Exemplo de utilização do software	107
	Apêndice C Tabelas do refinamento iterativo	140

Lista de Figuras

1.1	Método de Elementos Finitos	3
1.2	Erro de discretização entre MDF e MEF	4
2.1	Transferência de calor por condução entre dois corpos	12
2.2	Corrente convectiva natural	16
2.3	Transferência de calor por radiação entre dois corpos	17
2.4	Condições de fronteira dos três tipos excepto radiativa	19
2.5	Diferentes abordagens do MEF	20
2.6	Malha de dois elementos	25
3.1	Malha de elementos finitos e referenciais locais	32
3.2	Triângulo de Pascal	32
3.3	Polinómios de Chebyshev de grau 0 até 4	33
3.4	Mapeamento do triângulo de Pascal na matriz M	34
3.5	Assemblagem da matriz dos coeficientes	35
3.6	Matrizes constituintes do sistema	35
3.7	Utilização das estruturas topológicas na montagem do sistema	40
4.1	Fluxograma do programa desenvolvido	53
5.1	Caso de estudo 1 e malha utilizada na sua discretização	57
5.2	Campo de temperaturas e graus adoptados	58
5.3	Campos de fluxos esperado para o caso de estudo 1	58
5.4	Variação da energia térmica no processo iterativo	59
5.5	Norma do erro da energia no processo iterativo	60
5.6	Norma da densidade do erro na fronteira no processo iterativo	60
5.7	Graus finais das bases de aproximação	61
5.8	Campos de temperaturas	62
5.9	Campos de fluxos (esq q_x , dir q_y)	63
5.10	Comparação dos diferentes critérios com tolerância de selecção de 1%	64
5.11	Caso de estudo 2 e a malha utilizada na sua discretização	65
5.12	Campo de temperaturas esperado para o caso de estudo 2	65

5.13	Campos de fluxos esperados para o caso de estudo 2	66
5.14	Variação da energia com os graus de liberdade	66
5.15	Norma do erro da energia no processo iterativo	67
5.16	Norma da densidade do erro na fronteira no processo iterativo	67
5.17	Graus finais das bases de aproximação	68
5.18	Campos de temperaturas	69
5.19	Campos de fluxos (esq q_x , dir q_y)	70
5.20	Comparação dos três critérios com tolerância nula de selecção	72
5.21	Comparação dos três critérios com tolerância de selecção de 5%	72
5.22	Comparação dos três critérios com tolerância de selecção de 15%	73
5.23	Graus finais para diferentes critérios, tolerância de selecção nula	74
5.24	Graus finais para diferentes critérios, tolerância de selecção de 5%	74
5.25	Graus finais para diferentes critérios, tolerância de selecção de 15%	74
5.26	Caso de estudo 3 e a malha utilizada na sua discretização	75
5.27	Campo de Temperaturas esperado e graus utilizados para o caso de estudo 3	76
5.28	Campos de Fluxos esperados para o Caso de estudo 3	76
5.29	Variação da energia com graus de liberdade	77
5.30	Norma do erro da energia no processo iterativo	77
5.31	Norma da densidade do erro na fronteira no processo iterativo	78
5.32	Graus finais das bases de aproximação	78
5.33	Campos de temperaturas	79
5.34	Campos de fluxos (esq q_x ; dir q_y)	80
5.35	Campos de fluxos da iteração final em três dimensões	81
5.36	Comparação do caso de estudo 3 com tolerância de selecção de 1% e diferentes critérios	82
5.37	Graus finais das bases de aproximação com diferentes critérios	82
5.38	Campos de temperaturas obtidos com diferentes critérios	83
B.1	Menu inicial	95
B.2	Menu inicial - Dados geométricos	96
B.3	Menu inicial - Graus iniciais das bases de aproximação	96
B.4	Menu inicial - Condutividade térmica e aquecimento interno	97
B.5	Menu inicial - Integração numérica	97
B.6	Menu inicial - Botões Reset e Next	98
B.7	Malha gerada	98
B.8	Menu 2 - Definição das fronteiras exteriores	99
B.9	Menu 2 - Lista de fronteiras exteriores	99
B.10	Menu 2 - Definir tipo de fronteira	100

B.11 Menu 2 - Botões Previous, Reset e Next	100
B.12 Menu 3 - Definição das condições de fronteira	101
B.13 Menu 3 - Listas de fronteiras Dirichlet e Neumann	101
B.14 Menu 3 - Definição das temperaturas e fluxos de calor	102
B.15 Menu 3 - Botões Reset, Previous e Run	102
B.16 Malha - Botão de condições de fronteira	103
B.17 Exemplo de campo de temperaturas	103
B.18 Exemplo de campo de fluxos	104
B.19 Graus da malha resultante do refinamento.	104
B.20 Tabela com informações sobre cada iteração	105
B.21 Gráficos finais	105

Lista de Tabelas

3.1	Estruturas de dados utilizadas	39
C.1	Processo iterativo para o caso de estudo 1	109
C.2	Processo iterativo para o caso de estudo 2	110
C.3	Processo iterativo para o caso de estudo 3	113

Lista de Abreviaturas e Símbolos

Lista de Abreviaturas

MDF	Método das Diferenças Finitas
MEF	Método dos Elementos Finitos
GUI	Graphical User Interface ou Interface Gráfica
gdl	Graus de liberdade

Lista de Símbolos

Símbolos Gerais

Símbolo	Unidades	Descrição
A	m^2	área de secção
a_i	—	coeficientes do método dos resíduos pesados
L	m	comprimento da fronteira
σ	$\frac{W}{m^2 \times K^4}$	constante de Stefan-Boltzmann
$Dim(X)$	—	dimensão do vector X ou número de incógnitas estáticas
$Dim(Y)$	—	dimensão do vector Y ou número de incógnitas cinemáticas
x	m	distância entre dois pontos do mesmo corpo
Ω	—	domínio
U_i	—	energia (térmica) para a iteração i
q	$\frac{J}{m^2 \times s}$	fluxo de calor
q_x	$\frac{J}{m^2 \times s}$	fluxo de calor na direcção x
q_y	$\frac{J}{m^2 \times s}$	fluxo de calor na direcção y
q_z	$\frac{J}{m^2 \times s}$	fluxo de calor na direcção z
q_r	$\frac{J}{m^2}$	fluxo de calor radiante incidente por unidade de área
Γ_Ω	—	fronteira exterior total do domínio
Γ_u	—	fronteira de Dirichlet
Γ_σ	—	fronteira de Neumann
Γ_r	—	fronteira de Robin

Γ_{int}	—	fronteira interior
f_i	—	funções de teste
w_i	—	funções de peso no método de Galerkin
\dot{x}	—	gradiente da solução
n_D	—	grau ou ordem da base de aproximação no domínio
n_Γ	—	grau ou ordem da base de aproximação na fronteira
\cap	—	intersecção de conjuntos
n	—	normal exterior ao plano da superfície
α	—	número de indeterminação estática
β	—	número de indeterminação cinemática
N_D	—	número de termos presentes na base de aproximação no domínio
N_Γ	—	número de termos presentes na base de aproximação na fronteira
$\ \diamond\ $	—	módulo da norma de \diamond
∇	—	operador diferencial ou <i>nabla</i>
Q	$\frac{J}{s}$	quantidade de calor
R	—	resíduo
ϕ	—	solução para o método dos resíduos pesados
T	K ou $^{\circ}C$	temperatura
T_Γ	$^{\circ}C$	temperatura aplicada na fronteira de Dirichlet
t	s	tempo
\cup	—	união de conjuntos
ΔX	—	variação da solução
ε_Γ	—	indicador do erro da densidade do erro na fronteira
ε_V	—	indicador do erro da densidade do erro no domínio
$\varepsilon_{\Delta U}$	—	indicador do erro da variação de energia
$\varepsilon_{\Delta X}$	—	indicador do erro do incremento das temperaturas generalizadas

Matrizes e Vectores

Símbolo	Unidades	Descrição
\bar{q}	—	aproximação do fluxo de calor
\bar{T}	—	aproximação da temperatura
\mathbf{U}	—	base de aproximação no domínio
\mathbf{Z}	—	base de aproximação na fronteira
$\bar{\mathbf{U}}$	—	base de funções referente ao grau refinado no domínio
$\bar{\mathbf{Z}}$	—	base de funções referentes ao grau refinado na fronteira

\bar{t}_Γ	—	escalar referente ao refinamento em fronteira de Dirichlet
\mathbf{K}	—	matriz de condutividade
\mathbf{B}	—	matriz de equilíbrio das fronteiras
$\overline{\mathbf{K}}_1$	—	matriz referente aos novos graus de liberdade no domínio
$\overline{\mathbf{K}}$	—	matriz referente aos novos graus de liberdade no domínio
$\overline{\mathbf{B}}$	—	matriz de compatibilidade referente ao grau refinado
$\overline{\mathbf{X}}$	—	novos graus de liberdade no domínio
$\overline{\mathbf{Y}}$	—	novos graus de liberdade na fronteira
\mathbf{A}	—	submatriz do sistema refinado
\mathbf{B}^*	—	submatriz do sistema refinado
$\overline{\mathbf{C}}$	—	submatriz do sistema refinado
$\dot{\mathbf{q}}$	—	submatriz no refinamento do domínio
\diamond^T	—	transposta da matriz/vector \diamond
$\Delta \mathbf{Y}$	—	variação da solução devido a refinamento
\mathbf{W}_i	—	vector das funções de peso
\mathbf{X}	—	vector dos coeficientes de recuperação das temperaturas
\mathbf{b}	—	vector da geração interna de calor
\mathbf{q}_Γ	—	vector dos fluxos impostos nas fronteiras de Neumann
\mathbf{t}_Γ	—	vector das temperaturas impostas nas fronteiras de Dirichlet
\mathbf{Y}	—	vector dos coeficientes de recuperação dos fluxos
$\overline{\mathbf{q}}_\Gamma$	—	vector dos fluxos impostos devido ao refinamento no domínio
$\Delta \mathbf{x}$	—	vector da variação da solução

Características de Materiais

Símbolo	Unidades	Descrição
c	$\frac{\text{J}}{\text{kg} \times \text{K}}$	calor específico
h_s	$\frac{\text{W}}{\text{m}^2 \times \text{K}}$	coeficiente de transmissão convectivo
k	$\frac{\text{W}}{\text{m} \times \text{K}}$	condutividade térmica
α	$\frac{\text{m}^2}{\text{s}}$	difusividade térmica
ε	—	emissividade do corpo
ρ	$\frac{\text{kg}}{\text{m}^3}$	massa volúmica ou massa específica

Capítulo 1

Introdução

1.1 Motivação

A temperatura faz parte do nosso quotidiano. Medimo-la quando estamos doentes, ajustamo-la quando tomamos banho ou controlamo-la ao cozinhar, e desde sempre foi um factor fulcral no estudo científico da Natureza e seus processos. O que é, as suas consequências e como se transfere são as primeiras questões que colocamos quando pensamos em temperatura. A transferência de calor está presente em diversos campos da engenharia e o seu estudo é imperativo na resolução de vários problemas na vida humana. Desde situações banais de como aquecer ou arrefecer a nossa habitação até ao seu estudo complexo em viagens espaciais, a análise da propagação do calor é fundamental para a nossa qualidade de vida.

O desenvolvimento de ferramentas auxiliares como o computador alterou para sempre o curso da vida humana: um auxílio imprescindível para a evolução da sociedade durante o século XX, moldando a actual para o futuro, cheio de novos desafios e novos problemas.

Um passo lógico será o de aplicar estas capacidades a processos e métodos existentes envolvendo cálculo extensivo, de forma a torná-los mais rápidos e eficientes, facilitando assim a vida ao analista que estuda e trabalha com estes métodos. O crescimento, quase exponencial, da capacidade de processamento anuncia um futuro no qual o limite de computação vai ficar cada vez mais alargado. Problemas complexos e morosos são assim possíveis de resolver, com recurso a computadores, em questões de minutos, ao invés de horas ou dias.

Este trabalho propõe-se a contribuir para o desenvolvimento de ferramentas computacionais com o intuito de implementar, melhorar e aperfeiçoar métodos existentes com o auxílio do computador, desenvolvendo algoritmos de decisão, para de forma inteligente, rápida e eficiente, resolver o problema proposto.

Para o caso concreto de transferência de calor, as aplicações são inúmeras. Todo

e qualquer ramo de manufactura necessita de estudos de transferências térmicas nos seus produtos, pelo que um processo mais rápido e eficiente que execute esse estudo é muito vantajoso e rentável.

Diversos ramos da engenharia, mecânica, civil ou aeroespacial, entre outros, beneficiam bastante com um avanço tecnológico na área, sendo fulcral incluir investigações de transferência de calor no seu corpo de estudo, obtendo um quadro mais completo do comportamento e previsão do mesmo, para inúmeros cenários.

Um exemplo concreto na área da engenharia civil é o estudo de transferência de calor nas habitações no âmbito do conforto térmico e eficiência energética, factor com crescente importância fruto de um consumidor mais exigente e sensível a esta temática. Ainda dentro da área, outro caso seria, por exemplo, a aplicação do estudo da transferência de calor dentro de um edifício, em situações de incêndio, para a avaliação de risco de colapso devido ao fogo ou uma melhor colocação do sistema de aspersores, visto que as cabeças destes são activadas individualmente e por calor, e não por detecção de fumo como a maioria das pessoas julga. Uma distribuição mais eficiente, nas zonas críticas de transferência de calor aumentariam drasticamente a eficácia do sistema, diminuindo os riscos para as pessoas no local e minimizando os danos materiais resultantes.

1.2 Objectivos do Trabalho

Na presente dissertação, o objectivo pretendido é o de desenvolver, implementar e validar um modelo de simulação de transferência de calor, baseado na formulação híbrida do método dos elementos finitos (MEF), e aplicá-lo ao problema particular de transferência de calor por condução em regime estacionário.

Será também desenvolvido um algoritmo de refinamento p -adaptativo, detectando as zonas mais susceptíveis de ocorrer erros na solução, tanto em domínios como nas fronteiras, onde se vão refinar e enriquecer as respectivas bases de funções de aproximação. O processo de refinamento é feito de forma iterativa, garantindo-se que a solução apresenta convergência do ponto de vista energético, e obtendo, assim, uma solução optimizada.

O modelo de transferência de calor e o respectivo algoritmo de refinamento são testados em exemplos presentes na literatura estudada na preparação do trabalho, cuja solução analítica é conhecida, sendo desta forma avaliado o desempenho dos mesmos.

Uma interface gráfica é criada para a introdução de dados durante a fase de pré-processamento de forma a simplificar todo o processo para o utilizador menos familiarizado com a escrita em código computacional.

1.3 Estado da Arte

O desenvolvimento de métodos de elementos finitos para a resolução de problemas práticos da engenharia começou com o aparecimento do computador digital. A essência de uma solução de elementos finitos é a transformação de um problema definido por equações diferenciais num sistema ou conjunto de equações lineares algébricas, que posteriormente é resolvido. Este processo tira partido da capacidade computacional do computador digital, tornando-se muito eficaz e útil, sendo generalizado e aplicado a diferentes áreas da engenharia.

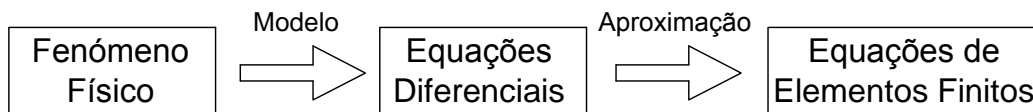


Figura 1.1: Método de Elementos Finitos

Antes de 1952, a análise estrutural era restrita a elementos ligados por apenas dois pontos no espaço. Engenheiros de estruturas usavam a analogia de treliça para modelar as suas estruturas. Contudo, esta analogia não podia ser aplicada para áreas não-rectangulares, aspecto comum na modelação de problemas mais complexos. Ray Clough e John Turner, no início da década de 50, enquanto empregados na Boeing, desenvolveram o trabalho pioneiro que impulsionaria a investigação na área dos elementos finitos [45].

Este foi um campo da engenharia muito trabalhado e aperfeiçoado nos anos 60, tendo um dos maiores pólos de estudo a nível mundial na Universidade de Berkeley, California, e mais tarde no Massachusetts Institute of Technology.

É difícil de localizar exactamente no tempo a sua criação, mas as raízes do método de elementos finitos podem ser traçadas a três áreas de investigação distintas: na matemática, Courant [16]; na física, Synge [42]; e na engenharia, Argyris e Kelsey [1]. Vários investigadores, principalmente na área da engenharia, contribuíram para a evolução gradual e partilha de conhecimentos, publicando uma série de artigos inovadores. Entre os autores acima mencionados, ocupam também lugares de destaque os trabalhos de Martin, Topp, Zienkiewicz, Cheung [9] e Bank [7].

Uma grande variedade de métodos numéricos tem sido utilizada na análise de problemas de transferência de calor. Alguns destes métodos eram usados antes do aparecimento do computador, e desde então tornaram-se redundantes ou obsoletos.

Os dois métodos que atingiram maior popularidade e sucesso desde o surgimento do computador digital têm sido o Método das Diferenças Finitas (MDF) e o Método dos Elementos Finitos (MEF). O MDF tem bastantes vantagens, principalmente a

sua simplicidade conceptual e facilidade de implementação. O método é aplicado directamente às equações diferenciais aproximando-as por expressões de diferenças apropriadas. No entanto, o principal factor que limita o MDF é a divisão do domínio em estudo através de fronteiras rectilíneas. Consequentemente, na análise de outros tipos de fronteira, a aproximação será feita num processo iterativo. O Método dos Elementos Finitos, por sua vez, não está restringido por esta limitação, e é até a sua versatilidade de se adaptar a domínios não regulares que o torna tão poderoso e útil. Ao contrário do MDF, o MEF não transforma as equações diferenciais em diferenças, mas divide o domínio em estudo em várias áreas simples que não se intersectam (elementos) de tamanhos variáveis (se necessário), dentro das quais a solução é aproximada utilizando funções de forma. As fronteiras destes elementos podem ser planares ou curvas. É apresentado na Figura 1.2, o problema típico do erro de discretização, comparando entre os dois métodos, onde o MEF geralmente tem vantagem. A geração de malha de elementos finitos pode ser um processo mo-

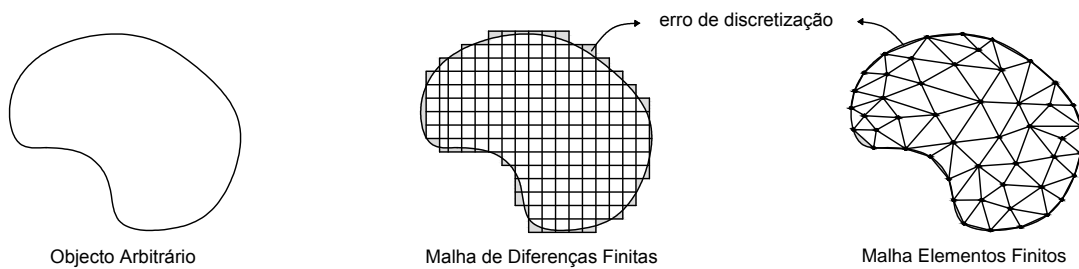


Figura 1.2: Erro de discretização entre MDF e MEF

roso e com grande tendência a erros. Isto é especialmente verdade em modelações computacionais de fenómenos reais, onde as malhas geralmente têm de ser geradas em geometrias tridimensionais com diferentes níveis de complexidade. Consequentemente, desde cedo na sua investigação foram criados geradores automáticos de malhas, que não são mais que algoritmos capazes de gerar malhas de elementos finitos válidas sobre domínios arbitrários, necessitando apenas da informação geométrica das fronteiras do domínio e as dimensões preponderantes dos seus elementos. Nesta área destaca-se o trabalho pioneiro, no início da década de 70, de Zienkiewicz e Phillips [48]. Desde então, muitas metodologias têm sido propostas e diferentes algoritmos desenvolvidos, como o desenvolvido pelo grupo de Análise de Estruturas do Departamento de Engenharia Civil do IST, nomeadamente o algoritmo de geração de malhas desenvolvido por Piteri [38], referido em [36].

A variável aproximada no domínio do elemento (temperatura, neste caso), é assumida que varie de acordo com as funções interpoladoras incluídas na base de funções do domínio. Por consequência, resultados de boa qualidade podem ser obtidos de duas maneiras, utilizando graus elevados nas bases de funções interpoladoras, ou

um elevado número de elementos finitos na discretização do domínio do problema. Devido a estas características, o MEF tem sido o método numérico mais explorado para a análise de problemas complexos de transferência de calor com resultados de precisão muito apreciável.

Está bem estabelecido na literatura que o uso de técnicas de refinamento adaptativo pode aumentar a confiança e precisão de soluções de elementos finitos. O objectivo de um procedimento adaptativo é o de melhorar automática e iterativamente a qualidade das soluções calculadas. Tipicamente, os procedimentos adaptativos funcionam redefinindo as malhas de forma contínua e reduzindo o erro de discretização até a solução convergir à precisão desejada pelo analista. Existe um corpo de trabalho extensivo na aplicação de técnicas adaptativas em problemas de elasticidade [30, 32, 50], dinâmica de fluidos [34, 35, 43, 47], e de transferência de calor [17, 21, 22, 26]. Uma resenha histórica do método é apresentada em [14, 45]. A literatura clássica para o tema do MEF, de uma forma mais abrangente, encontra-se em [8, 15, 39, 49].

Os elementos finitos podem ser classificados em duas formulações e suas respectivas sub-formulações. A primeira classe inclui os elementos finitos *convencionais*, sendo os mais utilizados na resolução de problemas e os mais comuns nos programas de elementos finitos comerciais. Estes elementos garantem, através da escolha das funções de aproximação no domínio dos elementos, que a solução obtida será ou localmente compatível, ou localmente equilibrada. Os elementos *não convencionais* [19], tipicamente menos utilizados, por outro lado, não fornecem soluções nem localmente compatíveis, nem localmente equilibradas, mas permitem flexibilidade na escolha das funções de aproximação, que podem ser diferentes em elementos diferentes (refinamento- p localizado).

As diferenças entre os dois tipos de elementos são significativas, como já foi estudado em [41]. Nos elementos finitos convencionais de temperatura, a continuidade das temperaturas entre os elementos é imposta localmente, adoptando-se funções pouco complexas, tipicamente lineares, enquanto que a continuidade dos fluxos é imposta em forma fraca. Assim, como a continuidade é imposta de maneira diferente, a qualidade da aproximação nos campos de temperatura e fluxos difere de acordo com o tipo de elemento empregue. Já com os elementos finitos não convencionais, tanto a continuidade das temperaturas como a dos fluxos de calor nas fronteiras dos elementos são impostas em forma fraca e a aproximação dos campos no domínio é feita a partir de funções mais complexas numericamente. Consequentemente a qualidade das soluções obtidas é mais equilibrada e obtêm-se melhores aproximações para os campos dos fluxos de calor.

De acordo com as restrições impostas às bases de aproximação, são definidas três sub-formulações dentro da formulação não convencional: *híbrida-mista*, *híbrida*

e *híbrida-Trefftz*. Como é referido por Freitas *et al* [19], o que distingue as formulações híbridas, híbridas-mistas ou híbridas-Trefftz é essencialmente as condições de campo (equilíbrio, compatibilidade ou elasticidade) que as funções de aproximação no domínio têm de assegurar localmente.

Para cada sub-formulação, dois modelos podem ser desenvolvidos, nomeadamente o modelo de deslocamentos e o modelo de tensão, ou temperatura e fluxo de calor respectivamente, no caso concreto de transferência de calor. Os modelos de temperatura e de fluxo de calor de qualquer formulação híbrida diferem fundamentalmente no que diz respeito à equação governativa localmente respeitada no domínio e à condição de fronteira explicitamente aplicada. O objectivo do desenvolvimento do modelo de temperatura é gerar soluções cinematicamente admissíveis, no domínio, e de modo alternativo, a utilização do modelo de fluxo de calor produz soluções estaticamente admissíveis no domínio do elemento.

As formulações não convencionais apresentadas aqui usam variáveis generalizadas em vez de variáveis nodais. Embora o conceito nodal, típico das formulações convencionais de elementos finitos, tenha sido expandido para os modelos híbridos, o seu uso limita, desnecessariamente, a escolha das funções de aproximação nas fronteiras essenciais dos elementos e, como tal, foi abandonado no decorrer deste trabalho.

A formulação híbrida-mista é a mais flexível das formulações híbridas em termos de escolha de bases de aproximação. De facto, os elementos híbridos e híbridos-Trefftz podem ser deduzidos do modelo híbrido-misto aplicando sucessivas restrições à base de domínio adoptada. A flexibilidade é a principal vantagem do método, pois permite virtualmente a escolha de qualquer base de aproximação, e desta forma tirar proveito de propriedades como ortogonalidade e/ou implementação computacional facilitada. O preço a pagar, contudo, é a necessidade de usar um número relativamente elevado de graus de liberdade de forma a atingir uma precisão razoável, pois normalmente as funções não têm qualquer relação com a real mecânica subjacente ao problema modelado. Isto tipicamente resulta na resolução de sistemas de grandes dimensões, ainda que esparsos.

A formulação híbrida pode ser deduzida directamente do modelo híbrido-misto correspondente, impondo que o campo das temperaturas (fluxos) satisfaça localmente a condição de compatibilidade (ou equilíbrio) no domínio dos elementos. Em termos de vantagens e desvantagens, o modelo híbrido é um compromisso entre a formulação híbrida-mista flexível e a formulação híbrida-Trefftz, mais restrita. A escolha das bases do domínio ainda é flexível, atingindo soluções analíticas para a maioria dos integrais, contudo a escolha de funções de aproximação é mais limitada. O número de graus de liberdade necessários para obter soluções adequadas é significativamente reduzido quando comparado com a formulação híbrida-mista. Por

estas razões, a formulação híbrida é a adotada neste trabalho, e será exposta de forma mais detalhada no Capítulo 2.

A formulação híbrida-Trefftz pode ser deduzida do modelo híbrido correspondente, garantindo que os campos de deslocamentos e fluxos satisfaçam localmente todas as equações do domínio. A maior vantagem da formulação híbrida-Trefftz é a sua intrínseca qualidade das funções de aproximação do domínio, que se relaciona com características físicas do fenómeno modelado. Como consequência directa desta propriedade, as expressões de todos os termos do sistema resolutivo são definidos por integrais de fronteira. Ademais, resultados precisos podem ser obtidos usando um número relativamente baixo de graus de liberdade e/ou malhas de elementos finitos pouco refinadas. Isto, por sua vez, conduz a sistemas resolventes relativamente pequenos. O preço a pagar, contudo, é a falta de flexibilidade na escolha das funções de aproximação, que geralmente, se tornam numericamente pesadas e impossíveis de integrar analiticamente. Vários exemplos de implementações usando esta formulação são apresentados em [20, 25, 26].

Para qualquer das formulações descritas, as variáveis associadas com as bases no domínio são estritamente dependentes do elemento, enquanto as bases na fronteira são partilhadas, no máximo por dois elementos ligados. Esta característica dota todas as formulações apresentadas de serem particularmente bem adequadas a processamento paralelo [32].

Por outro lado, existem várias razões para não optar imediatamente pelos elementos finitos não convencionais, tais como, complexidades conceptuais ou dificuldades no estabelecimento de bases de aproximação apropriadas.

Com efeito, o uso correcto do método dos elementos finitos, na formulação não convencional, necessita de conhecimento e sensibilidade que o utilizador inexperiente não possui. Esta formulação tem muitos parâmetros calibráveis que influenciam o seu desempenho, e que se forem mal escolhidos, podem comprometer a solução obtida, e devem, por isso, ser definidos criteriosamente. No entanto, novos avanços no campo do processamento paralelo [13] e contínuo melhoramento do equipamento informático renovam o interesse nestas formulações.

Três classes de métodos de refinamento foram desenvolvidos para melhorar a qualidade das soluções obtidas através do MEF [44]. A mais comum é o *refinamento h*, que se baseia no aumento da qualidade da malha através da redução da dimensão de alguns elementos finitos, mantendo a dimensão de outros. O segundo, e o abordado na presente dissertação, é o *refinamento p* [5], que mantém a malha inalterada, mas aumenta a ordem das funções de aproximação usadas no domínio e na fronteira [6]. Existe uma variante destes métodos, *refinamento h-p*, que é uma combinação do refinamento *p* e *h* [4]. A última classe, *refinamento r*, reorganiza a malha de elementos finitos, mantendo o mesmo número de elementos e de nós.

Quando o processo é iterativo e depende dos resultados prévios em várias etapas do cálculo este considera-se adaptativo. Tais métodos adaptativos foram inicialmente explorados por Babuška e Rheinboldt no final da década de 70 [2, 3]. Várias abordagens para atingir estimativas *a posteriori* do erro no sentido do processo adaptativo foram desenvolvidas na década de 80 [27, 28].

Aplicações modernas continuam a ser descobertas e exploradas, exemplos como simulações electromagnéticas ou modelação estocástica, para resolução de problemas probabilísticos são alguns dos exemplos recentes onde o MEF é empregue. O método dos elementos finitos ainda está em franca expansão e continua a ser uma área de investigação actual, surgindo novas áreas de interesse que fomentam o desenvolvimento do campo.

1.4 Organização do Trabalho

O texto apresentado de seguida é dividido em seis capítulos.

No segundo capítulo são introduzidos os conceitos teóricos sobre o tema em discussão. É apresentada a classificação dos tipos de transferência de calor, as respectivas equações governativas e condições de fronteira. A forma fraca dessas equações, obtida através do método dos resíduos pesados, é demonstrada neste capítulo, e com base nesta forma fraca, são derivadas as equações algébricas do modelo de temperatura dos elementos finitos híbridos, focado neste trabalho.

No capítulo três é abordada a implementação computacional, explicando de que forma é aplicada a metodologia geral para a análise de elementos finitos; como os dados são estruturados, o método de integração numérica utilizado e os processos de pré- e pós-processamento. A geração de malha, montagem do sistema e solução do mesmo são ainda discutidas.

No capítulo quatro, o algoritmo de refinamento p -adaptativo desenvolvido é explorado, primeiro expondo a estratégia de refinamento adoptada, nomeadamente, para os refinamentos na fronteira e no domínio. De seguida, o algoritmo é descrito na forma de pseudo-código e com o auxílio de um fluxograma, explicando a tomada de decisão imposta ao computador. Finalmente, aspectos de implementação do programa são abordados e as suas características explicadas.

No capítulo cinco, o desempenho dos elementos e da ferramenta de refinamento p -adaptativo implementados são analisados para três casos de estudo, obtendo diversa informação referente ao processo iterativo na forma de tabelas e gráficos. A solução obtida é processada e são montados os respectivos campos aproximados da temperatura e fluxos de calor, a cada passo do processo iterativo.

No capítulo seis, as conclusões são expostas e diferentes ideias de trabalhos futuros para o contínuo desenvolvimento desta área de estudo são propostas.

Em anexo, estão três apêndices, com informação auxiliar para a leitura do trabalho e melhor compreensão do mesmo. No apêndice A é descrita a dedução da forma fraca da equação governativa do problema de transferência de calor. No apêndice B é dado um exemplo de utilização do interface gráfico [23] (GUI) para um problema concreto de transferência de calor, para facilitar a utilização do programa desenvolvido. No apêndice C estão tabelados os processos iterativos dos casos de estudo analisados no presente trabalho.

Capítulo 2

Fundamentos Teóricos

Neste capítulo são apresentados os diferentes mecanismos de transferência de calor explicando como funcionam e de que forma são relevantes para o tema do presente trabalho, com especial foco no caso da condução. Os tipos de condições de fronteira que podem ocorrer associados a cada mecanismo são apresentados. O conceito de forma fraca de uma equação diferencial é discutido e particularizado para funções de ponderação de Galerkin, empregues neste trabalho. As diferentes formulações de elementos não-convencionais são introduzidas com particular ênfase na formulação híbrida, que será utilizada no presente trabalho.

2.1 Mecanismos de Transferência

Existem três mecanismos que contribuem para o processo de transferência de calor em diferentes proporções: *condução*, *convecção* e *radiação*. Condução é a transferência de calor pela vibração molecular em corpos sólidos. Convecção é a transferência de calor por movimento de fluido que ocorre por diferenças de densidade provocadas pela diferença de temperaturas entre as várias zonas do fluido. Este processo pode ocorrer *naturalmente* ou de forma *forçada* mediante acção de forças exteriores. Radiação é a transferência de energia electromagnética entre dois corpos a diferentes temperaturas.

É importante salientar que, quando se trata da teoria de transferência de calor, existe uma distinção entre transferência de calor no regime *estacionário* e transferência de calor no regime *não estacionário* ou *transitório*. O primeiro envolve campos de temperaturas e fluxos de calor constantes no tempo. O estado não-estacionário implica que as temperaturas e os fluxos de calor estão continuamente a ser alteradas, ou seja, dependentes do tempo. A transferência de calor no estado estacionário será o abordado no presente trabalho.

De seguida é explicada de maneira detalhada cada mecanismo, com especial ênfase na condução por ser a natureza do problema em estudo.

2.1.1 Condução

Transferência de energia por condução ocorre quando diferentes partes de um corpo sólido ou dois corpos em contacto estão a diferentes temperaturas. O fluxo de energia na forma de calor tem o sentido do mais quente, estado *mais* energético, ou seja com maior *agitação molecular*, para o mais frio, estado *menos* energético, com menor *agitação molecular*. Uma representação esquemática da transferência de calor por condução entre dois corpos é apresentada na Figura 2.1.

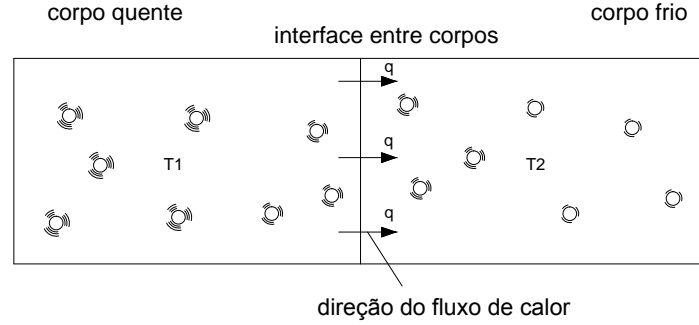


Figura 2.1: Transferência de calor por condução entre dois corpos

A quantidade de calor transferida entre duas superfícies verticais situadas a uma distância x , em condições estacionárias é dada por:

$$Q = k \frac{T_1 - T_2}{x} A t, \quad (2.1)$$

onde as variáveis presentes são,

Q - quantidade de calor (J ou Nm);

T - temperatura (K ou °C), com índices 1 e 2 referentes às duas partes do corpo;

t - tempo (s);

x - distância (em m) de separação entre as duas superfícies;

A - área da secção (m^2) para o fluxo de calor;

k - condutividade térmica ($Wm^{-1}K^{-1}$).

Derivando a equação (2.1) em relação ao tempo resulta na taxa de transferência de calor:

$$\frac{\partial Q}{\partial t} = k \frac{T_1 - T_2}{x} A \quad (2.2)$$

A equação (2.2) apenas é válida para problemas estacionários. Para problemas transitórios, a equação (2.2) pode ser redefinida na forma diferencial,

$$\frac{\partial Q}{\partial t} = -kA \frac{\partial T}{\partial x} \quad (2.3)$$

A equação (2.3) relaciona a quantidade de calor transferida por unidade de tempo $\frac{dQ}{dt}$ com o gradiente de temperatura do material $\frac{dT}{dx}$, e é conhecida como equação de condução de calor.

A quantidade de calor transferida por unidade de tempo através de uma área unitária é designada por fluxo de calor,

$$q = \frac{\frac{\partial Q}{\partial t}}{A} = -k \frac{\partial T}{\partial n} \left[\frac{J}{m^2 s} \right] \quad (2.4)$$

onde n é a normal exterior à superfície A . A condutividade térmica, k , pode ter um valor constante em todas as direcções, sendo neste caso o material designado por isotrópico, ou pode tomar a forma de uma matriz, com valores distintos para cada direcção, caso do material ortotrópico.

A equação (2.4) é a lei fundamental que rege a transmissão de calor por condução em materiais isotrópicos e é denominada *Lei de Fourier*, tendo sido proposta em 1822 pelo matemático francês, que postula que o calor transferido é proporcional ao gradiente de temperatura. O sinal negativo na equação foi introduzido para reflectir o significado físico do sentido da transferência de calor, nomeadamente da zona com temperatura mais alta para o com temperatura mais baixa.

Em geral, a área A não é ortogonal a nenhum dos eixos. Assim, os fluxos de calor são na realidade quantidades vectoriais. Num referencial cartesiano (x, y, z) são definidos da seguinte maneira:

$$q_x = -k \frac{\partial T}{\partial x} \quad (2.5)$$

$$q_y = -k \frac{\partial T}{\partial y} \quad (2.6)$$

$$q_z = -k \frac{\partial T}{\partial z} \quad (2.7)$$

ou na forma vectorial,

$$\mathbf{q} = -k \nabla T, \quad (2.8)$$

onde o ∇ é o operador gradiente, dado por:

$$\nabla = \left\{ \begin{array}{c} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{array} \right\} \quad (2.9)$$

Para materiais ortotrópicos, os fluxos na direcção de cada eixo, de acordo com a lei de Fourier, ficam

$$q_x = - \left(k_{11} \frac{\partial T}{\partial x} + k_{12} \frac{\partial T}{\partial y} + k_{13} \frac{\partial T}{\partial z} \right) \quad (2.10)$$

$$q_y = - \left(k_{21} \frac{\partial T}{\partial x} + k_{22} \frac{\partial T}{\partial y} + k_{23} \frac{\partial T}{\partial z} \right) \quad (2.11)$$

$$q_z = - \left(k_{31} \frac{\partial T}{\partial x} + k_{32} \frac{\partial T}{\partial y} + k_{33} \frac{\partial T}{\partial z} \right) \quad (2.12)$$

As equações que governam o problema de condução de calor em materiais isotrópicos, com condutividade térmica constante para todas as direcções, são agora definidas. Considerando um elemento de volume infinitesimal ($dx dy dz$) cujas faces são paralelas a três eixos ortogonais entre si, e sendo q_x , q_y e q_z os fluxos de calor nas direcções x , y e z , respectivamente, por unidade de tempo, a diferença entre o fluxo que sai e o fluxo que entra no elemento, Δq , é dada por

$$\begin{aligned} \Delta q = & dy dz \left(q_x + \frac{\partial q_x}{\partial x} dx - q_x \right) + dx dz \left(q_y + \frac{\partial q_y}{\partial y} dy - q_y \right) \\ & + dx dy \left(q_z + \frac{\partial q_z}{\partial z} dz - q_z \right) \end{aligned} \quad (2.13)$$

Para haver conservação da energia térmica, a soma do fluxo de calor que entra no elemento com o calor gerado internamente, $Q dx dy dz$, deve ser igual à soma da variação da energia interna com o fluxo de calor que sai do elemento. A variação da energia interna é dada pelo calor ganho no elemento na unidade de tempo, $\rho c \frac{\partial T}{\partial t} dx dy dz$, onde t é o tempo e ρ e c são a massa específica e o calor específico do material, respectivamente.

Fazendo o balanço energético obtém-se a equação diferencial da condução de calor,

$$\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} + \rho c \frac{\partial T}{\partial t} = Q \quad (2.14)$$

Substituindo-se os fluxos de calor nas direcções x , y e z na equação (2.14), obtém-se

$$\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + Q = \rho c \frac{\partial T}{\partial t} \quad (2.15)$$

ou na forma vectorial,

$$\nabla^T k \nabla T + Q = \rho c \frac{\partial T}{\partial t} \quad (2.16)$$

A equação (2.14) está escrita em termos de um sistema de coordenadas *Cartesianas* (x, y, z). Do ponto de vista prático, especialmente para análise numérica, pode ser útil utilizar o sistema de coordenadas *cilíndrico*. A equação (2.14), pode ser reformulada no sistema de coordenadas cilíndrico (não utilizado no presente trabalho), como:

$$\frac{1}{r} \frac{\partial}{\partial r} \left(r k \frac{\partial T}{\partial r} \right) + \frac{1}{r^2} \frac{\partial}{\partial \theta} \left(k \frac{\partial T}{\partial \theta} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + Q - \rho c \frac{\partial T}{\partial t} = 0 \quad (2.17)$$

A equação (2.15) pode ser reescrita após manipulação algébrica, na seguinte forma

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} + \frac{1}{k} Q = \frac{1}{\alpha} \frac{\partial T}{\partial t} \quad (2.18)$$

ou na forma vectorial,

$$\nabla^2 T + \frac{1}{k} Q = \frac{1}{\alpha} \frac{\partial T}{\partial t} \quad (2.19)$$

onde α é a *difusividade térmica*, $\frac{k}{\rho c}$ ($m^2 s^{-1}$) e ∇^2 é o operador de Laplace, dado por:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}. \quad (2.20)$$

O caso particular do regime não estacionário, sem geração interna de calor é escrito da seguinte forma

$$\rho c \frac{\partial T}{\partial t} = k \nabla^2 T \quad (2.21)$$

ou

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T, \quad (2.22)$$

sendo a equação (2.22) designada equação da difusão.

O significado físico desta propriedade é associado com a velocidade de propagação de calor no material sólido. Materiais com *altos valores de difusividade*, tais como metais, *aquecem mais rapidamente* que os materiais com baixos valores, como a madeira, por exemplo.

Considerando o caso do problema de condução de calor tridimensional num corpo isotrópico, com geração interna de calor, em regime não estacionário, a equação governativa do domínio é da forma:

$$\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + Q = \rho c \frac{\partial T}{\partial t} \quad (2.23)$$

Para o regime permanente, sendo $\frac{\partial T}{\partial t} = 0$, a equação (2.23) toma a forma:

$$\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + Q = 0 \quad (2.24)$$

2.1.2 Convecção

O processo de convecção natural envolve, inicialmente, a transferência de calor por condução de uma superfície de maior temperatura para as camadas adjacentes de fluido, resultando num decréscimo de densidade das mesmas, provocando a ascensão do fluido aquecido. Quando atinge o topo do volume onde está inserido, o movimento induzido pela diferença de densidades faz com que o mesmo se desloque para a periferia onde é costume que esteja significativamente mais frio e desça, sendo o fluido aquecido junto da superfície substituído por fluido em aquecimento proveniente da coluna central, dando continuidade à corrente convectiva. Uma representação esquemática da convecção natural é apresentada na Figura 2.2. Existe também a convecção forçada, e como o nome indica, ao contrário da natural, esta decorre de estímulos externos, não naturais. Qualquer mecanismo ou tipo de transporte resultante no movimento de fluido por acção de fontes externas, tais como bombas, ventoinhas, aparelhos de sucção, entre outros, é considerado um processo de convecção forçada. O caso de ar condicionado é um exemplo clássico disso.

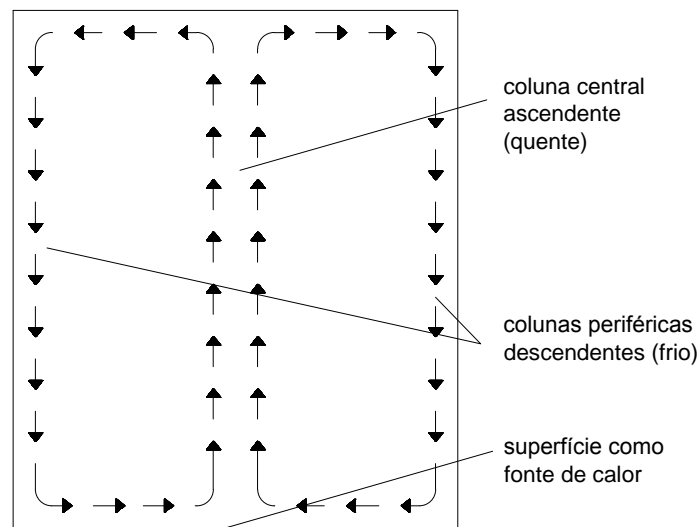


Figura 2.2: Corrente convectiva natural

A equação básica da transferência por convecção é:

$$Q = h_s A (T_b - T_s) \quad (2.25)$$

Nesta equação, Q é a quantidade de calor transferida (Js^{-1}), h_s o coeficiente de transmissão da superfície ($Wm^{-2}K^{-1}$), A a área da superfície (m^2), T_b a temperatura da superfície e T_s a temperatura média do fluido que a rodeia (K). Normalmente, o coeficiente de transmissão é determinado experimentalmente visto depender de vários factores de difícil previsão e alguma complexidade.

2.1.3 Radiação

A radiação térmica não necessita, ao contrário dos mecanismos de condução e convecção, de nenhum meio para transmitir energia, inclusive sendo mais eficiente no vácuo. Um exemplo comum e tangível de radiação térmica é a luz solar. Qualquer pessoa já sentiu o calor proveniente da luz solar na pele, e esta “viajou” pelo espaço que é um vácuo, até à Terra. A transferência de energia por radiação divide-se em três fases: primeira, a conversão da energia térmica de uma fonte quente para ondas electromagnéticas; segunda, o transporte destas ondas desde a fonte quente até ao receptor frio; e terceira, a absorção e reconversão das ondas electromagnéticas para energia térmica. Qualquer corpo com temperatura acima de $0K$ emite radiação térmica, logo ambos os corpos emitem e absorvem radiação térmica, mas o corpo de temperatura superior emite mais que o corpo de menor temperatura.

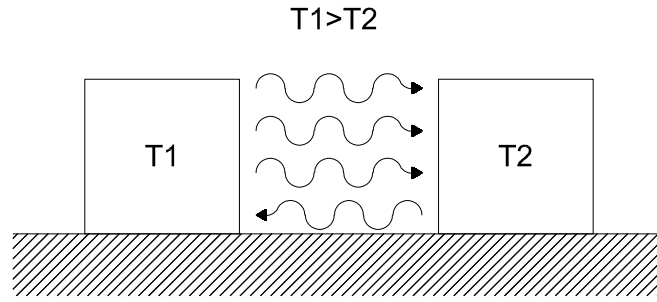


Figura 2.3: Transferência de calor por radiação entre dois corpos

A quantidade de energia radiada de uma superfície por unidade de tempo é designada por *potência de emissividade*, E . Para um irradiador perfeito, chamado *corpo negro*, a emissividade é proporcional à quarta potência da temperatura:

$$E = \sigma T^4 \quad (2.26)$$

onde σ é a *constante de Stefan-Boltzmann*, $5.670 \times 10^{-8} (Wm^{-2}K^{-4})$. A equação para a troca de calor por radiação entre os dois corpos a temperaturas T_1 e T_2 , respectivamente, é dada por:

$$Q = \varepsilon \sigma (T_1^4 - T_2^4) \quad (2.27)$$

onde Q é a diferença entre a energia por radiação que entra e que sai, por unidade de área, o ε é a *emissividade* (rácio entre a energia emitida por um *corpo cinzento* e um *corpo negro*). Um corpo cinzento é todo o corpo com emissividade inferior a 1 e superior a 0, e um corpo negro é um corpo que absorve toda a radiação electromagnética que nele incida. Para uma superfície muito polida ε pode ser tão baixo quanto 0.03, enquanto que para um corpo negro ε é 1.

2.2 Condições de fronteira

É assumido que as condições de fronteira podem ser dos seguintes tipos:

1. Temperatura específica

$$T_s = T_1(x, y, z, t) \quad (2.28)$$

2. Fluxo de calor específico, normal à fronteira

$$q_x n_x + q_y n_y + q_z n_z = -q_s \quad (2.29)$$

3. Convectiva

$$q_x n_x + q_y n_y + q_z n_z = h_s(T_b - T_s) \quad (2.30)$$

4. Radiativa

$$q_x n_x + q_y n_y + q_z n_z = \sigma \varepsilon T_s^4 - \alpha q_r \quad (2.31)$$

onde o fluxo de calor radiante incidente por unidade de área de superfície e as outras quantidades foram definidas na Secção 2.1.2. Para problemas transitórios é necessário especificar um campo de temperaturas inicial em cada ponto do corpo,

$$T(x, y, z, 0) = T_0(x, y, z). \quad (2.32)$$

A condição (2.28) é designada por condição de fronteira *Dirichlet* e é considerada de *primeiro tipo*. A mesma especifica o valor que a temperatura, T_r , deve ter ao longo da fronteira. A condição (2.29), chamada de condição de fronteira *Neumann*, é considerada de *segundo tipo*, e determina o valor da derivada normal, q_r , do campo de fluxos de calor na fronteira. A condição (2.30), denominada condição de fronteira *Robin*, é considerada de *terceiro tipo*, e é uma combinação ponderada de condições Dirichlet e Neumann, referente a condições convectivas na fronteira, q_r . A condição (2.31), de maior complexidade, é relativa a fenómenos de radiação e é frequentemente desprezada. É importante também mencionar que na eventualidade de existir uma fronteira radiativa, a condução passa a ser não linear.

A Figura 2.4 é uma representação esquemática das condições de fronteiras possíveis.

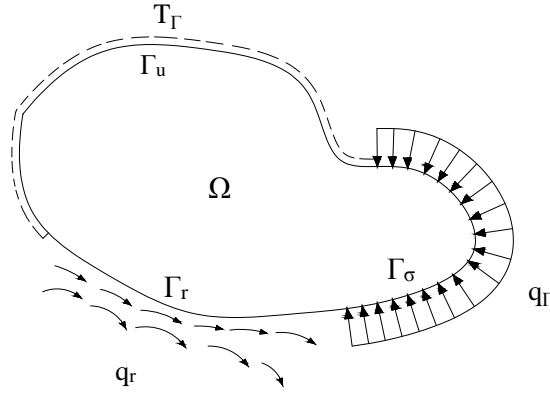


Figura 2.4: Condições de fronteira dos três tipos excepto radiativa

A área ocupada pela peça define o *domínio*, Ω , e a linha que a delimita define a *fronteira*, Γ_Ω . A fronteira pode ser discretizada em diferentes regiões, de acordo com a condição de fronteira imposta, sendo estas regiões complementares por não ser fisicamente possível impor num ponto, simultaneamente, uma temperatura e o fluxo de calor correspondente. Logo, a união de todas as fronteiras (fronteira de Dirichlet Γ_u , fronteira de Neumann Γ_σ e fronteira de Robin Γ_r) é a fronteira total do domínio, $\Gamma_\Omega = \Gamma_u \cup \Gamma_\sigma \cup \Gamma_r$ e estas nunca se intersectam em simultâneo, $\emptyset = \Gamma_u \cap \Gamma_\sigma \cap \Gamma_r$, ou entre si $\emptyset = \Gamma_u \cap \Gamma_\sigma$, $\emptyset = \Gamma_\sigma \cap \Gamma_r$ e $\emptyset = \Gamma_u \cap \Gamma_r$.

Na presente dissertação apenas as duas primeiras condições de fronteira são consideradas, a de Dirichlet e a de Neumann.

2.3 Método dos Resíduos Pesados

As formulações em elementos finitos podem ser derivadas a partir de diversas abordagens, como *princípios variacionais* ou *métodos de resíduos pesados*. O princípio variacional define uma expressão integral, chamada *funcional*, que inclui implicitamente as equações diferenciais e algumas das condições de fronteira de um problema. O princípio variacional assera tipicamente que a solução exacta do problema corresponde a um ponto de estacionaridade do funcional, geralmente um mínimo. O *Princípio da Energia Potencial Mínima*, com o objectivo de obter equilíbrios em sistemas mecânicos ou o *Princípio da Menor Dissipação* em viscosidade na dinâmica de fluidos, são exemplos de princípios variacionais.

Alternativamente, o Método dos Elementos Finitos pode ser derivado, utilizando o Método dos Resíduos Pesados (ou Ponderados) [29]. Essencialmente, este método impõe que as equações do problema sejam satisfeitas em média ponderada através de um conjunto arbitrário de funções. Obtém-se um conjunto de expressões integrais, conhecido como *forma fraca* do problema (por oposição à forma diferencial original, conhecida como *forma forte*). É importante referir que a solução da forma forte é sempre solução da forma fraca, ao passo que a afirmação contrária não é verdadeira.



Figura 2.5: Diferentes abordagens do MEF

Este método pode ser aplicado a qualquer problema regido por um sistema de equações diferenciais e condições de fronteira. Existem vários métodos de resíduos pesados, como o *método da colocação*, mas o mais popular e utilizado na presente dissertação, é o *método de Galerkin*.

O enunciado matemático de um problema físico regido por equações diferenciais é:

$$Du - f = 0 \quad (2.33)$$

para o domínio Ω .

Para o caso da condução de calor num material isotrópico com geração interna de calor, D é o operador Laplaciano ∇^2 , u é a temperatura T , e $f = -Q$. O problema escrito na *forma forte* (2.33), complementado pelas condições de fronteira apropriadas, implica que a equação diferencial tem de ser satisfeita em todos os pontos interiores e as condições de fronteira em todos os pontos das fronteiras.

Poucos problemas têm solução analítica que satisfaça a forma forte e, em geral, procura-se uma solução aproximada. Como é uma solução aproximada, a equação diferencial original não será satisfeita exactamente. A diferença entre a solução aproximada e a solução exacta chama-se residual R . Os métodos de resíduos pesados tentam minimizar o residual numa forma média ou em pontos específicos.

Consequentemente, como a solução aproximada \tilde{u} não satisfaz a equação (2.33) em todos os pontos, logo, o resíduo $R = R(x)$ fica:

$$R = D\tilde{u} - f \quad (2.34)$$

sendo \tilde{u} uma combinação linear das funções de base de aproximação. Tipicamente \tilde{u} é um polinómio de n termos cujo termo i é multiplicado por um coeficiente a_i que é um grau de liberdade generalizado. Os valores dos coeficientes são escolhidos de modo a minimizar o resíduo R . Estes valores são os melhores quando satisfazem a expressão seguinte de equações governantes na forma fraca.

$$\int W_i R dV = 0 \quad (2.35)$$

para $i = 1, 2, \dots, n$.

Onde cada $W_i = W_i(x)$ é uma função de ‘peso’. No método de Galerkin, cada função W_i é o multiplicador do correspondente termo a_i em \tilde{u} .

A premissa básica para o método dos elementos finitos é a de assumir uma solução na forma de:

$$\phi = \sum_{i=1}^n a_i f_i \quad (2.36)$$

onde f_i são as *funções de teste* e a_i os coeficientes a ser determinados na solução. Uma *função de teste*, *função interpoladora* ou *função de forma* são diferentes formas de designar estas funções. O que difere de método para método é como se definem estas funções. Interpolar consiste em definir uma função contínua que satisfaça condições estabelecidas para um número finito de pontos. Na análise de elementos finitos, estes pontos são tipicamente os nós de um elemento, e a condições atribuídas são valores nodais de um campo (temperaturas, no problema concreto da presente dissertação). Os valores nodais são raramente exactos, e mesmo quando são, a interpolação geralmente providencia valores aproximados noutros pontos.

Método da colocação

No método da colocação, adoptam-se para as funções de teste a função delta de Dirac, tal que

$$W_j(x, y) = \delta_j(x, y) = \begin{cases} 1 & \text{se } (x, y) = (x_j, y_j) \\ 0 & \text{se } (x, y) \neq (x_j, y_j) \end{cases} \quad (2.37)$$

Tendo em conta que

$$\int_{\Omega} \delta_j(x, y) f(x, y) d\Omega = f(x_j, y_j) \quad (2.38)$$

esta condição corresponde a impôr que o resíduo é nulo em pontos discretos do domínio, chamados pontos de colocação, onde a equação diferencial governativa é satisfeita.

Método de Galerkin

A particularidade do método de Galerkin é a escolha das funções das bases de aproximação como funções interpoladoras, tomando partido das propriedades que as caracterizam. Este método conduz ao aparecimento de matrizes simétricas: uma

vantagem do ponto de vista computacional. Nesta técnica os coeficientes a_i são determinados das n equações:

$$\int_{\Omega} f_i R d\Omega = 0 \quad (2.39)$$

para $i = 1, 2, \dots, n$.

onde Ω é o domínio da solução.

2.4 Formulação em elementos híbridos

Está bem estabelecido na literatura referida no Capítulo 1 que a formulação não convencional compara favoravelmente em termos de sensibilidade à distorção da malha e incompressibilidade, com taxas de convergência e precisão de resultados superiores em relação à formulação convencional.

Para além do relativo alto número de graus de liberdade que envolve, ainda que mais ou menos comparável com elementos convencionais, a maior vulnerabilidade desta formulação é o risco reduzido que tem para a geração de modos espúrios, que mais à frente no texto são explicados. O grande número de graus de liberdade é compensado pela estrutura altamente esparsa do sistema resolvente, e a facilidade de resolução que tal implica. Actualmente já existem ferramentas muito eficientes para armazenamento, manipulação e resolução, de forma directa ou iterativa, de grandes sistemas, altamente esparsos. Mais ainda, estes sistemas são adequados a processamento paralelo em qualquer das fases da análise e resolução do problema, porque as suas variáveis são dependentes de um só elemento e, no máximo, duas fronteiras.

Os elementos finitos híbridos, para problemas de condução térmica podem ser formulados utilizando o modelo de temperaturas ou o modelo de fluxos. O modelo de temperaturas, utilizado neste trabalho obriga a satisfação, à partida, da equação de Fourier no domínio de cada elemento, sendo que a equação da condução de calor é imposta em forma fraca. Pelo contrário, o modelo de fluxos obriga a escolher uma aproximação do campo de fluxos que satisfaça a equação de condução de calor e impõe em forma fraca a lei de Fourier. As condições de fronteira são impostas em forma fraca nas duas formulações.

2.4.1 Equações do modelo de temperaturas da formulação híbrida

O modelo de temperaturas da formulação híbrida baseia-se na aproximação directa do campo da temperatura no domínio de cada elemento, determinada por

$$\bar{T}(x, y) = \sum_{i=1}^{N_D} U_i(x, y) X_i = \mathbf{U}(x, y) \mathbf{X} \quad (2.40)$$

onde N_D é o número de termos presentes na base de aproximação do domínio, (\mathbf{U}) , e \mathbf{X} é o vector dos pesos correspondentes.

Para além da aproximação (2.40), o fluxo normal de calor é aproximado independentemente nas fronteiras de Dirichlet e interiores:

$$\mathbf{n}\bar{q}(x, y) = \sum_{j=1}^{N_\Gamma} Z_j(x, y) Y_j = \mathbf{Z}(x, y) \mathbf{Y} \quad (2.41)$$

onde N_Γ é o número de termos presentes na base de aproximação (\mathbf{Z}) nas fronteiras essenciais, e \mathbf{Y} é o vector dos graus de liberdades correspondentes. As únicas condições impostas às bases \mathbf{U} e \mathbf{Z} é que sejam completas e linearmente independentes. É importante referir que, caso seja necessário, é possível adicionar soluções singulares à base de aproximação para problemas cuja modelação assim necessite. Isso pode aumentar a convergência da solução em problemas envolvendo concentrações de fluxos de calor ou temperaturas que, embora sejam situações invulgares em problemas fisicamente plausíveis, ocasionalmente podem ocorrer.

A escolha de adoptar estas bases implica que se perde o significado físico das soluções \mathbf{X} que se tinha nos elementos finitos convencionais. Por outro lado, as aproximações no domínio são independentes dos nós, permitindo uma considerável liberdade na selecção da geometria e refinamento dos elementos. Ao contrário dos elementos convencionais, é possível definir elementos com números arbitrários de nós e arestas e ainda escolher refinamentos diferentes para as bases de cada um.

A expressão da equação governativa do problema na forma fraca, deduzida no Apêndice A, tem a forma

$$\int_{\Omega} (\nabla w)^T k \nabla \bar{T} d\Omega + \int_{\Gamma_u} w \mathbf{n}\bar{q} d\Gamma_u + \int_{\Gamma_\sigma} w q_\Gamma d\Gamma_\sigma - \int_{\Omega} w Q d\Omega = 0 \quad (2.42)$$

Aplicando o Método de Galerkin, considere-se as funções de forma como 'pesos', $w \equiv U$, e substituindo as definições (2.40) e (2.41) na forma fraca (2.42), obtém-se:

$$\int_{\Omega} (\nabla U)^T k \nabla U d\Omega \mathbf{X} + \int_{\Gamma_u} \mathbf{U}^T \mathbf{Z} d\Gamma_u \mathbf{Y} + \int_{\Gamma_\sigma} \mathbf{U}^T q_\Gamma d\Gamma_\sigma - \int_{\Omega} \mathbf{U}^T Q d\Omega = \mathbf{0} \quad (2.43)$$

A equação anterior pode ser escrita, para um elemento, na forma

$$\mathbf{K} \mathbf{X} + \mathbf{B} \mathbf{Y} = \mathbf{b} - \mathbf{q}_\Gamma \quad (2.44)$$

com

$$\mathbf{K} = \int_{\Omega} (\nabla \mathbf{U})^T k \nabla \mathbf{U} d\Omega = \int_{\Omega} \left(\frac{\partial \mathbf{U}^T}{\partial x} k \frac{\partial \mathbf{U}}{\partial x} + \frac{\partial \mathbf{U}^T}{\partial y} k \frac{\partial \mathbf{U}}{\partial y} \right) d\Omega \quad (2.45)$$

$$\mathbf{B} = \int_{\Gamma_u} \mathbf{U}^T \mathbf{Z} d\Gamma_u \quad (2.46)$$

$$\mathbf{b} = \int_{\Omega} \mathbf{U}^T Q d\Omega \quad (2.47)$$

$$\mathbf{q}_{\Gamma} = \int_{\Gamma_{\sigma}} \mathbf{U}^T q_{\Gamma} d\Gamma_{\sigma} \quad (2.48)$$

Nos elementos híbridos, de maneira a garantir a continuidade das temperaturas nas fronteiras, a condição de fronteira de Dirichlet é imposta na forma fraca. Para um problema estacionário, a condição de fronteira de Dirichlet, sendo a temperatura T_{Γ} aplicada na fronteira, é dada por,

$$T = T_{\Gamma} \quad (2.49)$$

ou

$$T - T_{\Gamma} = 0 \quad (2.50)$$

e usando a base de aproximação \mathbf{Z} como função de teste, logo a condição de continuidade será expressa assim

$$\int_{\Gamma_u} \mathbf{Z}^T (T - T_{\Gamma}) d\Gamma_u = \mathbf{0} \quad (2.51)$$

Substituindo a definição (2.40) na equação anterior fica-se com:

$$\int_{\Gamma_u} \mathbf{Z}^T \mathbf{U} d\Gamma_u \mathbf{X} = \int_{\Gamma_u} \mathbf{Z}^T T_{\Gamma} d\Gamma_u \quad (2.52)$$

que pode ser re-escrito na forma matricial,

$$\mathbf{B}^T \mathbf{X} = \mathbf{t}_{\Gamma} \quad (2.53)$$

com,

$$\mathbf{t}_{\Gamma} = \int_{\Gamma_u} \mathbf{Z}^T T_{\Gamma} d\Gamma_u \quad (2.54)$$

No caso de haver mais de um elemento, como esquematizado na Figura 2.6, a continuidade de temperatura na fronteira interelementar é assegurada por uma condição do tipo (2.51).

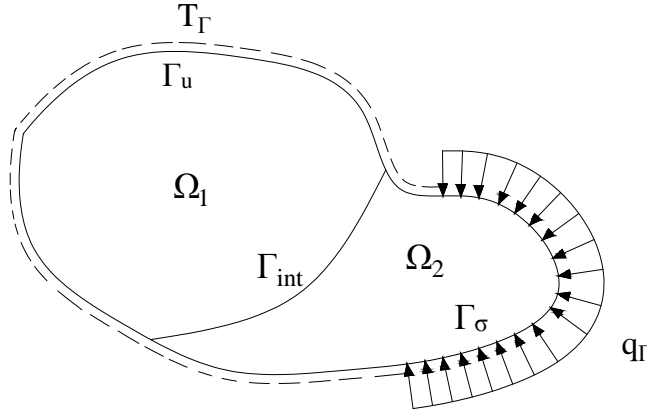


Figura 2.6: Malha de dois elementos

Considerando uma fronteira Γ_{int} partilhada por dois elementos, um dos elementos com temperatura T_1 e o outro elemento com temperatura T_2 , a continuidade entre elementos é garantida impondo que

$$\int_{\Gamma_{int}} \mathbf{Z}^T (T_1 - T_2) d\Gamma_{int} = \mathbf{0} \quad (2.55)$$

Substituindo a definição (2.40) na equação anterior fica-se com:

$$\int_{\Gamma_{int}} \mathbf{Z}^T \mathbf{U}_1 d\Gamma_{int} \mathbf{X}_1 - \int_{\Gamma_{int}} \mathbf{Z}^T \mathbf{U}_2 d\Gamma_{int} \mathbf{X}_2 = \mathbf{0} \quad (2.56)$$

ou

$$\mathbf{B}_1^T \mathbf{X}_1 - \mathbf{B}_2^T \mathbf{X}_2 = \mathbf{0} \quad (2.57)$$

A partir das equações (2.44), (2.53) e (2.57) consegue-se construir o sistema resolutivo global de elementos finitos híbridos,

$$\begin{bmatrix} \mathbf{K} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{O} \end{bmatrix} \begin{Bmatrix} \mathbf{X} \\ \mathbf{Y} \end{Bmatrix} = \begin{Bmatrix} \mathbf{b} - \mathbf{q}_\Gamma \\ \mathbf{t}_\Gamma \end{Bmatrix} \quad (2.58)$$

O sistema é simétrico e esparsa, a matriz de condutividade \mathbf{K} é diagonal por blocos, onde cada bloco é a matriz de condutividade de um elemento da malha de elementos finitos, não ocorrendo o somatório de matrizes típico da formulação de elementos convencionais. As matrizes \mathbf{B} são definidas nas fronteiras e estão associadas ao vector de fluxos partilhado por elementos adjacentes. O vector \mathbf{X} , de temperaturas generalizadas, depende apenas do respectivo elemento ao qual está associado. O vector \mathbf{Y} , dos fluxos generalizados, é no máximo partilhado por dois elementos, com uma fronteira interelementar comum.

As aproximações são desligadas dos nós e sendo independentes entre si, permitem a escolha de bases diferentes para elementos diferentes, promovendo uma maior

adaptabilidade e convergência num processo de refinamento. Por esta razão, o refinamento localizado pelo meio de adição de funções de aproximação locais é uma possibilidade. As bases são definidas hierarquicamente, com um grau polinomial associado, pelo que o refinamento p é feito aproveitando o processamento da iteração anterior, sem perda de dados. Finalmente, a malha de elementos finitos gerada pode ser conforme ou não conforme, aspecto indiferente para a formulação híbrida.

2.4.2 Indeterminação estática e cinemática

Para além da independência linear, a escolha das bases de aproximação nos domínios e nas fronteiras deve garantir que o problema seja, ao mesmo tempo, estática e cinematicamente indeterminado. A indeterminação estática impõe que os graus de liberdade estáticos incluídos no vector \mathbf{Y} não sejam determináveis apenas pelas equações de equilíbrio (primeiro conjunto de equações do sistema).

A indeterminação cinemática requer que os graus de liberdade cinemáticos presentes no vector \mathbf{X} não sejam determinados pelas equações de compatibilidade (o segundo conjunto de equações do sistema).

O valor de indeterminação estática α é dado pela diferença entre o número de incógnitas que constam da equação de equilíbrio e o número de equações respectivo. O valor de indeterminação cinemática β é dado pela diferença entre o número de incógnitas cinemáticas e o número de equações respectivo. As dimensões dos vectores \mathbf{X} e \mathbf{Y} são os números de incógnitas cinemáticas e estáticas respectivamente, logo os valores de α e β serão

$$\alpha = [\text{Dim}(\mathbf{X}) + \text{Dim}(\mathbf{Y})] - \text{Dim}(\mathbf{X}) = \text{Dim}(\mathbf{Y}) \geq 0$$

$$\beta = \text{Dim}(\mathbf{X}) - \text{Dim}(\mathbf{Y})$$

Pela definição, conclui-se que o sistema só é isoestático ($\alpha = 0$) caso a dimensão do vector \mathbf{Y} seja nula, logo, para uma malha de apenas um elemento e com nenhuma das suas fronteiras exteriores do tipo Dirichlet. Em qualquer outro caso, a indeterminação estática é garantida. A indeterminação cinemática é verificada se $\text{Dim}(\mathbf{X}) > \text{Dim}(\mathbf{Y})$, o que influenciará a escolha dos graus das bases de funções de aproximação. Como foi demonstrado em [41], adoptando funções de grau inferior na base de fronteira às da base do domínio consegue-se assegurar a indeterminação cinemática. No entanto, o quão inferior o grau tem de ser, deve ser cautelosamente decidido. Demonstrou-se num caso específico em [41] que apenas um grau de diferença, apesar de garantir a indeterminação cinemática, correspondia a um condicionamento excessivo dos graus de liberdade cinemáticos pelas condições de fronteira, proporcionando resultados de fraca qualidade.

Por outro lado, de forma a evitar inconsistências nas condições de compatibilidade e para garantir uma forte continuidade aquando da construção dos campos de temperatura entre elementos e na fronteira Dirichlet, os valores de β não devem ser excessivamente elevados, como é referido em [24]. Se a malha considerada for de vários elementos, com várias fronteiras interelementares, as funções presentes nas bases de aproximação passam a ser partilhadas entre elementos, o que relaxa a condição.

Capítulo 3

Implementação Computacional

Neste capítulo é apresentada a metodologia aplicada para a implementação computacional dos elementos finitos híbridos. A implementação foi feita de raiz, procurando-se garantir a maior flexibilidade na selecção dos refinamentos das bases de funções de aproximação e, para esse efeito, adoptou-se uma estrutura de dados topológica.

As geometrias e dimensões dos elementos adoptados e a geração de malha, as funções de forma escolhidas para as bases de aproximação no domínio e na fronteira, a montagem do sistema resolvente a partir das equações descritas no capítulo anterior e sua solução, assim como os métodos adoptados para a integração numérica são descritos neste capítulo.

Por fim, a estrutura de dados adoptada é descrita e exemplificada para um problema concreto.

3.1 Metodologia geral

A análise de um problema pelo Método dos Elementos Finitos tem geralmente três fases, que são, por ordem sequencial: a *fase de pré-processamento*, em que se recolhem e armazenam os dados sobre a estrutura e as acções que lhe são aplicadas, e se define a discretização da malha e o tipo de elemento empregue; a *fase de processamento*, em que se calculam os coeficientes do sistema resolvente que é de seguida resolvido, determinando a solução do problema; e a *fase de pós-processamento*, em que se caracteriza a resposta da estrutura à acção imposta, através da representação, por norma gráfica, dos campos das soluções do problema. Esses podem ser campos de deslocamento e de tensão, no caso de problemas de análise estrutural, e campos de temperatura e de fluxo, no caso de problemas de transferência de calor.

A resolução em elementos finitos de um problema segue uma metodologia padrão, constando dos seguintes passos:

1. Definir o problema a ser resolvido em termos de equações diferenciais. Cons-

truir a forma integral do problema como trabalho virtual, princípio variacional ou formulação fraca.

2. Definir o tipo dos elementos finitos a ser usados na análise, e as respectivas funções de aproximação.
3. Definir a malha para o problema. Isto envolve a descrição do arranjo dos nós e elementos, tal como a especificação das condições de fronteira e outros parâmetros para a formulação utilizada.
4. Calcular os termos do sistema resolutivo, utilizando as expressões obtidas através da formulação integral do problema e montar o sistema.
5. Resolver o sistema e obter os pesos das funções de aproximação.
6. Construção das aproximações dos campos nos domínios e nas fronteiras e representação gráfica dos mesmos.

3.2 Hipóteses e simplificações

Um número de importantes suposições foram feitas no decorrer deste trabalho. A condição primária é que as partículas do material estão em repouso, considerando condução de calor em sólidos e estruturas. Caso a análise de transferência de calor ocorresse em fluidos dinâmicos, seria necessário incluir um termo de transferência de calor convectiva sobre o fluido.

Outra simplificação assumida é que as condições de transferência de calor podem ser analisadas desacopladas das condições de tensões, por elas provocadas. Esta hipótese é válida em muitas análises estruturais, mas pode ser inadequada, por exemplo, para materiais como metais onde a sua deformação pode gerar calor e alterar os campos de temperatura. Tal alteração pode, por sua vez, afectar as propriedades do material e resultar em subsequentes deformações adicionais.

A última suposição importante adoptada é que não há alterações de fase, pelo que efeitos de calor latente são desprezados.

3.3 Implementação

A implementação do método divide-se nos seguintes tópicos:

- Geometria e geração de malha;
- Montagem do sistema;
- Solução do sistema.

3.3.1 Geometria e geração de malha

A geração de malha é automática e assegurada pelo programa aquando da introdução de dados pelo utilizador. Actualmente, as malhas são simples e regulares, quadrangulares ou rectangulares, o trabalho focando-se mais no refinamento das bases de aproximação.

Os nós, tipo de fronteira, graus das funções de aproximação e outras informações sobre a malha são definidos pelo utilizador e registados nas estruturas de dados do programa.

Os elementos e as fronteiras usam referenciais locais independentes. O referencial do elemento, consta de um sistema de eixos ortogonais com origem no centro do elemento e os eixos orientados nas direcções principais de inércia do elemento.

Os referenciais definidos nas fronteiras têm apenas um eixo, coincidente com a fronteira e que toma valores entre -1 e 1 em cada extremidade respectivamente, no sentido da fronteira, e 0 no centro da mesma. Para ilustrar a definição dos referenciais locais, a Figura 3.1 dá o exemplo de uma malha de 16 elementos. O mesmo procedimento é usado para qualquer número de elementos.

As condições de fronteira são introduzidas, na fase do pré-processamento, pelo utilizador. Tanto as condições de Dirichlet, que são temperaturas impostas, como as condições de Neumann, ou fluxos impostos, são definidos como funções polinomiais do grau que o utilizador desejar. Por exemplo, se as condições de fronteira forem definidas através de dois valores, o programa toma estes valores como sendo os valores dos campos nas extremidades da fronteira, num perfil linear. Por sua vez se se especificarem três valores, o programa já assume um perfil quadrático para o fluxo, com o primeiro e terceiro valor aplicados nos extremos e o segundo colocado no centro da fronteira. O mesmo raciocínio é extrapolado para qualquer introdução de valores.

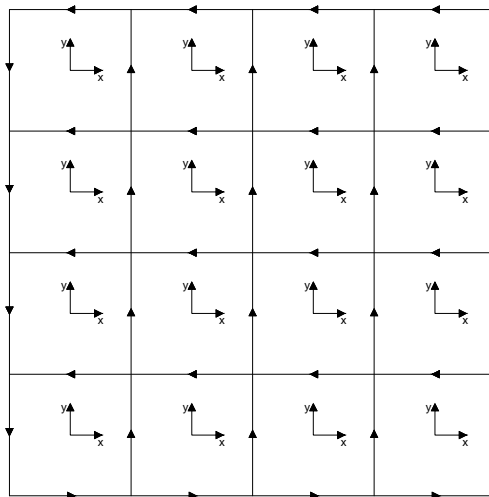


Figura 3.1: Malha de elementos finitos e referenciais locais

3.3.2 Funções de forma

A escolha do grau das funções de forma pode ter um impacto considerável no comportamento da aproximação polinomial desejada, como foi demonstrado em [40]. Na formulação híbrida, para montar o campo de temperaturas, são adoptadas bases independentes para o domínio de cada elemento, e para cada uma das fronteiras essenciais. As bases no domínio são compostas por monómios retirados do *triângulo de Pascal*, representado na Figura 3.2. As bases para as fronteiras, são construídas utilizando *polinómios Chebyshev*, representados na Figura 3.3.

A única condição que se tem de verificar é que ambas as bases de aproximação sejam *completas* e *linearmente independentes*. Na base de funções para o domínio, por forma a garantir que a base é sempre completa, são sempre adoptados graus completos do triângulo de Pascal, sendo cada grau de maior dimensão que o precedente.

grau	monómios					
0	1					
1	x		y			
2	x ²		xy	y ²		
3	x ³	x ² y	xy ²	y ³		
4	x ⁴	x ³ y	x ² y ²	x ² y ³	y ⁴	
5	x ⁵	x ⁴ y	x ³ y ²	x ² y ³	x ³ y ⁴	y ⁵
...	...					

Figura 3.2: Triângulo de Pascal

Os polinômios de Chebyshev, presentes na base de aproximação nas fronteiras, são uma família de polinômios que pode ser definida *recursivamente*. Assim, conhecendo os dois primeiros polinômios Chebyshev de primeiro tipo, $T_0(x)$ e $T_1(x)$, todos os outros polinômios $T_n(x)$, $n \geq 2$ podem ser obtidos através da fórmula de recorrência:

$$T_0(x) = 1 \quad (3.1)$$

$$T_1 = x \quad (3.2)$$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \quad (3.3)$$

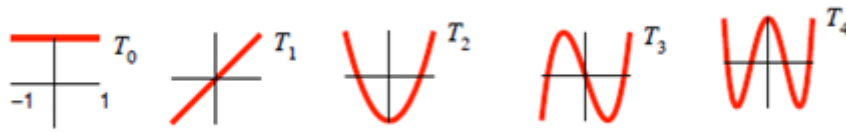


Figura 3.3: Polinômios de Chebyshev de grau 0 até 4

Em alternativa, o polinômio de Chebyshev de grau n pode ser obtido através da relação trigonométrica:

$$T_n = \cos[n \arccos(x)] = \cosh[n \operatorname{arccosh}(x)] \quad \text{para } n = 0, 1, 2, 3 \dots \quad (3.4)$$

Aproximação de funções por polinômios de Chebyshev

Os polinômios de Chebyshev são bastante úteis em métodos numéricos para a interpolação de funções no intervalo $-1 \leq x \leq 1$ porque:

- $|T_n(x)| \leq 1$ para o intervalo $-1 \leq x \leq 1$;
- Os máximos e mínimos são de ordens de grandeza semelhante;
- Os máximos e mínimos são espalhados de forma uniforme pelo intervalo $-1 \leq x \leq 1$;
- Potenciam um melhor condicionamento do sistema resolutivo;
- São fáceis de calcular e de converter de e para uma série de potências.

O conjunto destas propriedades produz uma aproximação polinomial que minimiza o erro de aproximação [18].

3.3.3 Montagem do Sistema

Como o programa não utiliza linguagem simbólica, a construção dos diferentes blocos do sistema ocorre a partir das bases de funções de aproximação no domínio (\mathbf{U}) e na fronteira (\mathbf{Z}). No caso da base no domínio, esta é montada a partir de uma matriz ‘mapa’ que listam os índices que permitem construir a base de funções polinomiais completa para um dado grau. O triângulo da base, representado na Figura 3.4a de forma esquemática, a título de exemplo, permite a construção de uma base no domínio de grau 5, como foi apresentada na Figura 3.2. A matriz de mapeamento, grava para cada termo da base, a linha e a coluna do mesmo no triângulo de Pascal, no formato $\binom{m}{n}$. Estes números, m para a linha e n para a coluna, são essenciais para a construção da base. Os dois índices, m e n permitem determinar os expoentes de cada monómio, através da fórmula $x^{m-n} \times y^n$ sendo a primeira linha grau $m = 0$. A primeira linha da matriz de mapeamento, representada na Figura 3.4b, guarda os índices m e a segunda linha os índices n , e cada coluna representa um termo resultante na base. De forma a ser completa, a base de um certo grau é o conjunto de todos os termos de graus inferiores ou iguais ao grau definido, existindo assim uma simetria em x e y , sem dar primazia a nenhuma das coordenadas.

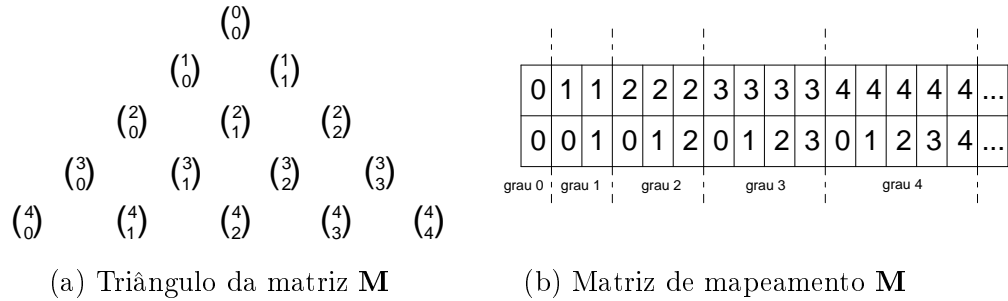


Figura 3.4: Mapeamento do triângulo de Pascal na matriz \mathbf{M}

Nas fronteiras, as bases de aproximação nas fronteiras são construídas a partir da definição trigonométrica (3.4) dos polinómios de Chebyshev.

Para ilustrar a construção do sistema, considera-se o exemplo prático, representado na Figura 3.5a, de uma malha com 4 elementos e 12 fronteiras e com as bases de aproximação de grau 3 no domínio e 1 na fronteira. A numeração a azul identifica os elementos e os números a verde as fronteiras. Para esta situação, a matriz dos coeficientes do sistema resolutivo encontra-se representado esquematicamente na Figura 3.5b, com os pontos azuis a representar entradas não nulas e a branco os elementos nulos, sendo perceptível a alta esparsidade da mesma, característica já mencionada na Secção 2.4.

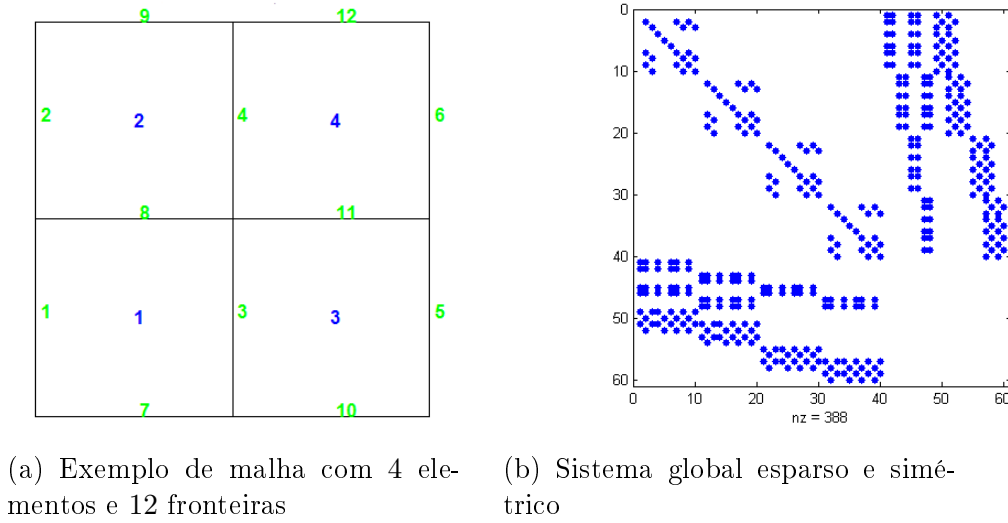


Figura 3.5: Assemblagem da matriz dos coeficientes

A matriz de mapeamento \mathbf{M} é utilizada para a construção das matrizes de condutividade \mathbf{K} , uma por cada elemento presente na malha, e delimitadas a vermelho na figura seguinte. Estas matrizes são quadradas e simétricas e têm a dimensão da base de funções de aproximação no domínio do elemento correspondente.

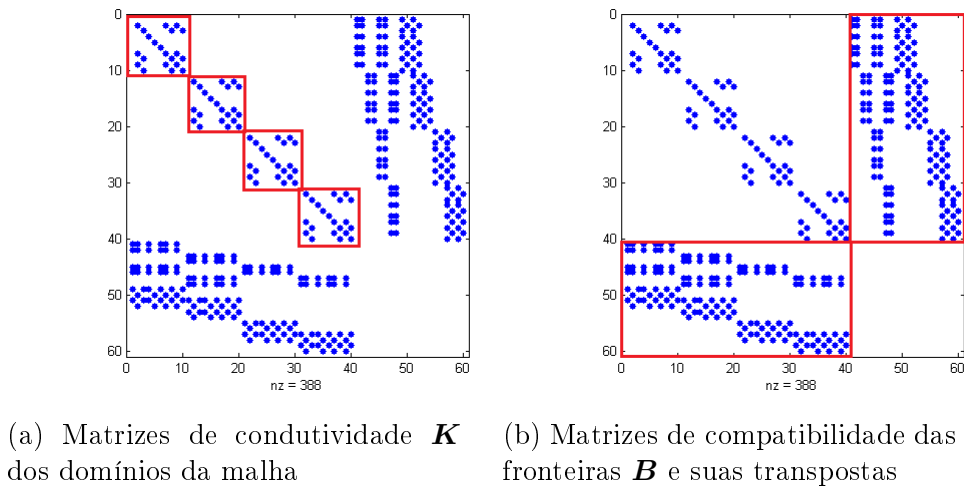


Figura 3.6: Matrizes constituintes do sistema

A partir das bases de funções de aproximação da fronteira, são criadas e montadas as matrizes \mathbf{B} , e as suas transpostas, no sistema global na forma representada a vermelho na Figura 3.6b. Caso a matriz \mathbf{B} pertença a uma fronteira Dirichlet *externa*, a mesma tem um número de linhas igual à dimensão da base no domínio ao qual a fronteira pertence e um número de colunas dado pela dimensão da base da fronteira respectiva. Caso seja uma fronteira *interna*, estando ligada a dois elementos, existirão duas matrizes nessa coluna, nas posições dos domínios respectivos.

3.3.4 Campos de Soluções

A solução do sistema (2.58) fornece uma estimativa única para o campo de temperaturas, tanto no domínio como nas fronteiras. Este campo é obtido a partir da definição (2.40) referida no capítulo anterior, logo

$$\begin{aligned} \bar{T}(x, y) = \mathbf{U}(x, y) \mathbf{X} &= \left\{ 1 \ x \ y \ x^2 \ xy \ y^2 \ \dots \right\} \begin{Bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ \vdots \end{Bmatrix} \\ &= X_1 + xX_2 + yX_3 + x^2X_4 + xyX_5 + y^2X_6 \dots \end{aligned} \quad (3.5)$$

No domínio há uma estimativa única do campo dos fluxos de calor, calculada da aproximação do campo de temperaturas, pela lei de Fourier (2.4),

$$\bar{\mathbf{q}}(x, y) = -k [\nabla \mathbf{U}(x, y)] \mathbf{X} = -k \begin{bmatrix} 0 & 1 & 0 & 2x & y & 0 & \dots \\ 0 & 0 & 1 & 0 & x & 2y & \dots \end{bmatrix} \begin{Bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ \vdots \end{Bmatrix} \quad (3.6)$$

O fluxo de calor no domínio para a direcção x pode ser obtido de

$$\bar{q}_x(x, y) = -k (X_2 + 2xX_4 + yX_5 + \dots) \quad (3.7)$$

e para a direcção y por

$$\bar{q}_y(x, y) = -k (X_3 + xX_5 + 2yX_6 + \dots) \quad (3.8)$$

Pelo contrário, nas fronteiras interiores e de Dirichlet, obtém-se uma estimativa múltipla do campo dos fluxos de calor normais, os mesmos podendo ser calculados a partir do vector $\mathbf{Y}^{(\Gamma)}$ de cada fronteira Γ , na forma

$$\bar{q}^{(\Gamma)}(x, y) = \mathbf{Z}^{(\Gamma)} \mathbf{Y}^{(\Gamma)} = \left\{ \begin{matrix} 1 & x & 2x^2 - 1 & 4x^3 - 3x & \dots \end{matrix} \right\} \left\{ \begin{matrix} Y_1^{(\Gamma)} \\ Y_2^{(\Gamma)} \\ Y_3^{(\Gamma)} \\ Y_4^{(\Gamma)} \\ \vdots \end{matrix} \right\} \quad (3.9)$$

$$= Y_1^{(\Gamma)} + (x) Y_2^{(\Gamma)} + (2x^2 - 1) Y_3^{(\Gamma)} + (4x^3 - 3x) Y_4^{(\Gamma)} + \dots$$

ou a partir do vector \mathbf{X} , na forma

$$\bar{q}^{\Gamma}(x, y) = - \{ n_x \ n_y \} k [\nabla U(x, y)] \mathbf{X} \quad (3.10)$$

Quando se trata de fronteiras interiores, os fluxos de calor nas fronteiras podem ainda ser calculados pela relação (3.10), a partir dos dois elementos que partilham a fronteira. A comparação entre as várias estimativas poderá permitir uma avaliação local do erro da aproximação, mas a ideia não é explorada neste trabalho.

3.3.5 Integração Numérica

O cálculo de todos os integrais que definem os termos do sistema resolutivo dos elementos híbridos é feito recorrendo a uma técnica de integração numérica baseada no *método de Gauss*.

O integral de uma função é aproximado como uma soma ponderada de valores da função em pontos específicos dentro do domínio de integração. A integração de Gauss é adequada para domínios de integração regulares de uma, duas ou três dimensões.

No método dos elementos finitos, a integração de Gauss é usualmente preferida devido ao seu melhor desempenho relativamente a outras fórmulas, como por exemplo a de Newton-Cotes, por necessitar de avaliar as funções em menos pontos [8]. Ela pode ser aplicada em casos onde uma função integral pode ser avaliada em pontos arbitrários dentro do intervalo de integração. Visto não existir nenhuma dificuldade em atingir estes requisitos em algoritmos de elementos finitos, a integração gaussiana é a regra de integração mais utilizada neste contexto.

A *regra de quadratura de Gauss*, nome em homenagem a Carl Frierich Gauss, é uma regra de quadratura construída para produzir um resultado exacto para polinómios de grau $2n - 1$ ou menor, por uma escolha apropriada de pontos x_i e pesos w_i para $i = 1, 2, \dots, n$.

O domínio de integração para a regra é, convencionalmente, escolhido para o intervalo $[-1, 1]$, pelo que a regra se enuncia da seguinte forma:

$$\int_{-1}^1 f(x)dx \approx \sum_{i=1}^n w_i f(x_i). \quad (3.11)$$

A quadratura Gaussiana produzirá resultados precisos desde que a função $f(x)$ seja bem aproximada pela função polinomial dentro do intervalo $[-1, 1]$.

3.4 Estrutura de Dados Topológica

Para se ter uma definição completa da geometria de um objecto é necessário um conjunto de informações essenciais que definam a sua forma e a localização espacial de todos os seus elementos constituintes. A topologia é a abstracção desse conceito e pode ser definida como o conjunto coerente de informações obtidas a partir da descrição geométrica completa de um objecto. Estas informações são armazenadas e estruturadas de acordo com uma lógica que garanta a consistência das relações, neste caso espaciais, como os nós pertencentes a uma fronteira ou fronteiras pertencentes a um elemento, formando estruturas de dados topológicas com a informação necessária para definir o objecto. Desta forma podem-se obter, de forma simples e rápida, informações relevantes aos elementos topológicos sem a necessidade de se fazer uma consulta global à geometria do modelo, fornecendo uma abordagem eficiente para a manipulação da informação.

Nas estruturas de dados não topológicas algumas das limitações são, por exemplo, a duplicação de informação sem controlo (por exemplo, as coordenadas globais de uma fronteira são comuns a dois ou mais elementos) ou não possibilitar a realização eficiente de operações envolvendo relacionamentos topológicos.

Como é referido em [11], esta opção é apropriada para processamento paralelo, facto suportado pelas características inerentes do sistema, concretamente, a estrutura esparsa e em blocos que não se sobrepõem [12]. O principal objectivo da organização nesta estrutura em paralelo é o de potenciar ao máximo a eficiência e rapidez no processamento e distribuição de informação entre processadores do sistema. O desempenho de uma eventual implementação paralela com vários processadores, é dependente do quão eficaz é a troca de informações entre os processadores do sistema. Neste trabalho tal processamento não é implementado, mas é, no entanto, adoptada uma estrutura de dados topológica para armazenar e aceder a informação de forma mais fluida.

O número de operações envolvidas, por exemplo, em:

- encontrar as fronteiras ligadas a um nó específico;
- encontrar os nós/fronteiras de um certo elemento;
- encontrar os elementos adjacentes a um dado elemento,

é constante ou varia linearmente com o número de entradas localmente envolvidas.

A generalidade, a robustez e a versatilidade são as características mais relevantes destas estruturas de dados, as quais foram aproveitadas durante a implementação para a organizar a informação envolvida. A memória necessária para armazenar cada membro destas estruturas é constante e cada operação que as envolve tem sempre um baixo custo, mesmo para malhas muito grandes. O desenvolvimento procura manter a generalidade, por forma a permitir a extensão a outros problemas, por exemplo à análise adaptativa ou materiais com comportamento não-linear.

Os dados necessários para manipular toda a informação aquando do processo de construção e reconstrução do sistema global durante o processo iterativo, é guardada e acedida de forma explícita em estruturas de dados. A informação referente aos elementos é armazenada na estrutura de dados **Loops**. A informação sobre as fronteiras é armazenada na estrutura de dados **Edges**. A informação sobre as condições de fronteira é armazenada na estrutura de dados **BConds**. As estruturas encontram-se listadas na Tabela 3.1.

Tabela 3.1: Estruturas de dados utilizadas

Loops Structure	Edges Structure	BConds Structure
Loops.nodes	Edges.nini	BConds.Dirichlet
Loops.edges	Edges.nfin	BConds.Neumann
Loops.center	Edges.parametric	
Loops.area	Edges.lleft	
Loops.order	Edges.lright	
Loops.dim	Edges.type	
Loops.insert	Edges.order	
	Edges.dim	
	Edges.insert	

No Apêndice B é listado de forma mais detelhada o que cada estrutura guarda e para que é utilizada.

Um exemplo prático da utilização deste tipo de estruturas é demonstrado na montagem do sistema descrito na Secção 3.3.3. Cada bloco é montado, de forma independente, na matriz global a partir das coordenadas da posição de entrada que são guardadas nas subestruturas **insert** pertencentes às estruturas **Loops** e **Edges**.

As dimensões de cada bloco, que na prática depende do grau da base de aproximação, são armazenadas na subestrutura `dim`.

Na Figura 3.7, o processo de montagem do sistema está esquematizado para os dois primeiros elementos e duas primeiras fronteiras numeradas.

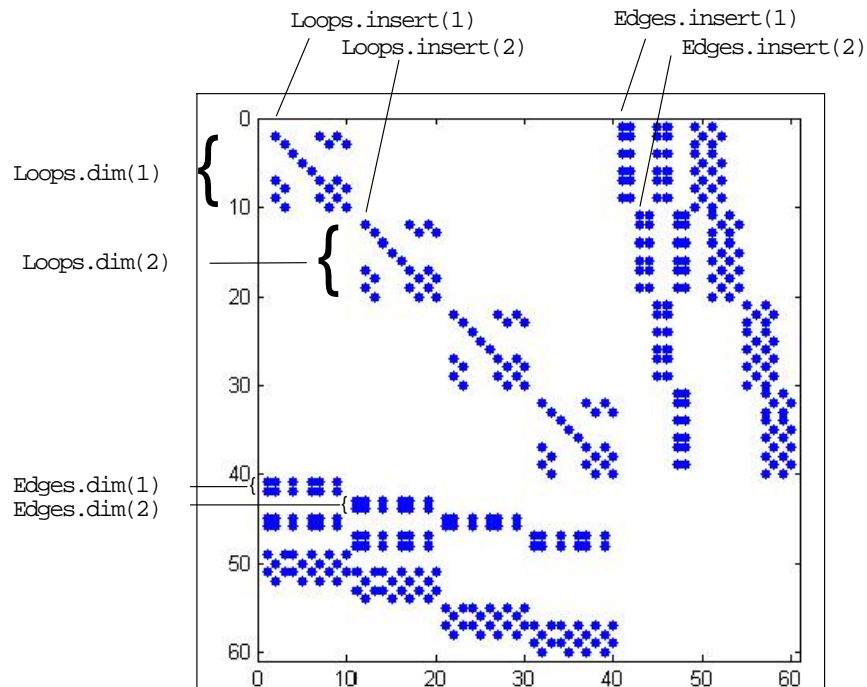


Figura 3.7: Utilização das estruturas topológicas na montagem do sistema

Capítulo 4

Refinamento p -Adaptativo

No presente capítulo é apresentado o algoritmo de refinamento p desenvolvido. Os objectivos do algoritmo são introduzidos e os diferentes tipos de erros resultantes dos cálculos realizados descritos. A estratégia de refinamento assumida é exposta e os passos do algoritmo sucintamente descritos e apresentados na forma esquemática de um fluxograma. É abordada a convergência esperada do processo iterativo e são brevemente referenciados aspectos relacionados com a implementação numérica do algoritmo p -adaptativo.

4.1 Aspectos gerais

Uma maneira de melhorar a qualidade de uma solução de elementos finitos será o de refinar as bases de aproximação dos elementos finitos, de uma maneira uniforme ou de forma adaptativa. A adaptatividade provém do facto do algoritmo identificar quais as fronteiras e elementos cujo refinamento localizado promovam uma convergência da solução, em vez de um acréscimo generalizado de todos os elementos e/ou fronteiras.

Uma das vantagens, previamente já mencionadas, da utilização da formulação em elementos finitos híbridos é flexibilidade na escolha do tipo de funções nas bases de aproximação e o seu grau, e ainda, que este refinamento pode ser feito de forma independente entre elementos e fronteiras conferindo-lhe maior versatilidade. O aumento do número de funções no domínio e/ou fronteiras não significa, necessariamente, uma melhoria na solução e pode, inclusive, piorar os resultados noutra região, pelo que é necessário garantir que os graus respeitem determinadas condições, como a indeterminação cinemática.

Esta flexibilidade da formulação híbrida pode ser considerada uma vantagem quando empregue por analistas experientes com o método, no entanto, para o utilizador menos familiarizado acaba por ser uma desvantagem pois pode acabar com uma distribuição de bases de aproximação mal equilibrada comprometendo a solu-

ção. Esta é uma das razões principais para a preferência em elementos convencionais nos programas comerciais de elementos finitos sobre os elementos híbridos.

O algoritmo de refinamento deve, de acordo com os critérios que lhe são implementados, manter este equilíbrio entre as aproximações (ou os graus das bases, de forma indirecta), escolhendo sempre as fronteiras e/ou domínios que apresentem o maior ‘erro’. Assim, é assegurado um refinamento criterioso, onde o aumento do grau da base de aproximação contribua para a convergência da solução.

A automatização do algoritmo de refinamento permite então aliviar a responsabilidade que o analista tem na escolha dos graus, determinante para um utilizador inexperiente, transferindo-a para o programa que decide, de acordo com o algoritmo implementado, que graus tomar. O algoritmo deve ser cuidadosamente programado, com todos os parâmetros que influenciem o processo de refinamento considerados, e os resultados obtidos escrutinados com espírito crítico.

A implementação por blocos na diagonal principal da matriz permite adicionar e remover polinómios nas bases de aproximação de qualquer elemento de forma eficiente e independente [46], guardando a estrutura matricial previamente definida na iteração anterior, sendo reestruturada na iteração actual com o acréscimo de graus de liberdade desejado.

O algoritmo foi implementado em ambiente MATLAB R2013a [31] para facilitar a manipulação de informação e desenvolver estruturas de dados, permitindo ao mesmo tempo a organização modular do programa. As vantagens deste tipo de abordagem encontra-se detalhada em [33].

O valor de algoritmos adaptativos é enaltecido no contexto de uma interface gráfica, permitindo ao utilizador acompanhar visualmente, as alterações efectuadas a cada ciclo no campo da solução e a convergência do método.

4.1.1 Origem do erro

Geralmente, as soluções obtidas pelo método dos elementos finitos para um problema físico são diferentes da realidade física, sendo esta diferença entre os valores obtidos (aproximados) e exactos (reais), o *erro*. O erro pode ocorrer por várias razões, e pode ser classificado segundo [37], em erro de *modelação*, erro de *discretização do domínio*, erro de *discretização* ou escolha das *funções de aproximação* e erro *numérico*.

O erro de modelação acontece caso o modelo matemático adoptado não simule de forma fidedigna o problema físico real cuja resolução o método dos elementos finitos pretende resolver. Esta construção virtual da realidade não reproduz, regra geral, o comportamento real do modelo físico devido a várias causas: as propriedades do material são aproximações das reais ou as condições de fronteira são diferentes da realidade, ou as próprias equações do modelo não tomam em conta todos os efeitos

relevantes à modelação do problema.

O erro de discretização do domínio, como o nome indica, depende de como o domínio é dividido. Um refinamento da malha adequado dos elementos finitos utilizados é fundamental para uma solução válida do problema.

O erro de discretização das funções é devido ao facto de, em geral, as funções de aproximação escolhidas para as bases de aproximação nos elementos finitos não conterem a solução exacta do modelo matemático adoptado. No caso dos elementos híbridos, as funções de aproximação dos elementos não permitam ainda recuperar exactamente as condições de fronteira, sejam essenciais ou naturais.

O erro numérico existe por causa dos erros inerentes a todos os métodos numéricos. Se for utilizada a integração numérica, muito provavelmente irá introduzir erros nos coeficientes do sistema resolutivo. Como os cálculos são efectuados em precisão finita, haverá erros de truncatura. Com o aumento de graus de liberdade, estes erros de truncatura também aumentam, e ficam mais proeminentes, ganhando relevância sobre erros de outras origens.

4.1.2 Estratégia de refinamento

Como já foi referido na secção anterior, o objectivo de um algoritmo de refinamento p -adaptativo é de obter a sequência ‘óptima’ de combinações de graus de liberdade de cada domínio (elemento) e cada fronteira que assegure que a solução tenha uma dada precisão.

Os aspectos essenciais deste procedimento são a selecção do critério que influencia o processo de tomada de decisão e a implementação computacional deste, na qual deve ser simples e computacionalmente económica, de forma a conferir uma convergência rápida e estável.

No caso dos elementos finitos que geram soluções localmente compatíveis ou equilibradas, a energia térmica tem uma convergência monótona com o número de graus de liberdade, que a maximização/minimização da mesma constitui o critério predilecto para estimar a qualidade de uma solução [41, 8, 49].

Contudo, é conhecido que o refinamento p de soluções produzidas por modelos de elementos híbridos não induz, geralmente, convergências monótonas de energia térmica. Este comportamento é demonstrado em [10], e é uma consequência do facto da formulação híbrida não produzir, tipicamente, soluções cinemática ou estaticamente admissíveis.

Uma vez que \mathbf{X} é uma variável local e \mathbf{Y} é partilhado no máximo por dois elementos será apresentada separadamente o processo de refinamento na fronteira e no domínio.

4.1.3 Refinamento na fronteira

O refinamento na fronteira é o aumento do número de termos N_Γ presentes na base de aproximação, ou graus de liberdade estáticos no sistema resolvente (2.58). É obtido através da incrementação de funções de aproximação reunidas na base \mathbf{Z} .

Considera-se que as funções de aproximação dos fluxos numa fronteira foram enriquecidas com um novo grau de liberdade \bar{Y} . O incremento na base de aproximação passa a ser definida como:

$$n\bar{q} = \mathbf{Z}(\mathbf{Y} + \Delta\mathbf{Y}) + \bar{\mathbf{Z}} \bar{Y} \quad (4.1)$$

Este novo grau de liberdade produz uma variação $\Delta\mathbf{Y}$ na solução corrente \mathbf{Y} para as fronteiras, e uma variação $\Delta\mathbf{X}$ na solução \mathbf{X} para os domínios. Uma nova equação de compatibilidade é adicionada ao sistema (2.58) tomando a forma em (4.2). A matriz $\bar{\mathbf{B}}$ é construída a partir da equação (2.46) utilizando a base $\bar{\mathbf{Z}}$ associada ao grau de liberdade adicionado que contém apenas a função do grau de liberdade acrescentado à base completa.

A matriz $\bar{\mathbf{B}}$ tem todas as entradas nulas, excepto as associadas aos elementos (no máximo dois), ligados ao lado enriquecido. Pela definição (2.54), note-se que \bar{t} é escalar no caso de se apenas adicionar um grau por refinamento e se usar a base de aproximação com funções de Chebyshev, caso contrário seria um vector. Este é nulo, quando o incremento ocorre numa fronteira exterior do tipo Dirichlet, na qual estejam impostas temperaturas nulas.

$$\begin{bmatrix} \mathbf{K} & \mathbf{B} & \bar{\mathbf{B}} \\ \mathbf{B}^T & \mathbf{O} & \mathbf{O} \\ \bar{\mathbf{B}}^T & \mathbf{O} & \mathbf{O} \end{bmatrix} \begin{Bmatrix} \mathbf{X} + \Delta\mathbf{X} \\ \mathbf{Y} + \Delta\mathbf{Y} \\ \bar{Y} \end{Bmatrix} = \begin{Bmatrix} \mathbf{b} - \mathbf{q}_\Gamma \\ \mathbf{t}_\Gamma \\ \bar{t}_\Gamma \end{Bmatrix} \quad (4.2)$$

O sistema (4.2) pode ser reescrito na seguinte forma,

$$\begin{cases} \mathbf{K}\mathbf{X} + \mathbf{K}\Delta\mathbf{X} + \mathbf{B}\mathbf{Y} + \mathbf{B}\Delta\mathbf{Y} + \bar{\mathbf{B}}\bar{Y} = \mathbf{b} - \mathbf{q}_\Gamma \\ \mathbf{B}^T\mathbf{X} + \mathbf{B}^T\Delta\mathbf{X} = \mathbf{t}_\Gamma \\ \bar{\mathbf{B}}^T\mathbf{X} + \bar{\mathbf{B}}^T\Delta\mathbf{X} = \bar{t}_\Gamma \end{cases} \quad (4.3)$$

onde resolvendo apenas para a variação imposta à solução e tendo em conta que \mathbf{X} e \mathbf{Y} satisfazem a equação (2.58), resulta no seguinte conjunto de equações,

$$\begin{cases} \mathbf{K}\Delta\mathbf{X} + \mathbf{B}\Delta\mathbf{Y} + \bar{\mathbf{B}}\bar{Y} = \mathbf{0} \\ \mathbf{B}^T\Delta\mathbf{X} = \mathbf{0} \\ \bar{\mathbf{B}}^T\mathbf{X} + \bar{\mathbf{B}}^T\Delta\mathbf{X} = \bar{t}_\Gamma \end{cases} \quad (4.4)$$

que pode ser expresso na forma matricial:

$$\begin{bmatrix} \mathbf{K} & \mathbf{B} & \overline{\mathbf{B}} \\ \mathbf{B}^T & \mathbf{O} & \mathbf{O} \\ \overline{\mathbf{B}}^T & \mathbf{O} & \mathbf{O} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{X} \\ \Delta \mathbf{Y} \\ \overline{\mathbf{Y}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{O} \\ \mathbf{O} \\ \bar{t}_\Gamma - \overline{\mathbf{B}}^T \mathbf{X} \end{Bmatrix} \quad (4.5)$$

Este novo sistema pode ser simplificado, utilizando as seguintes notações,

$$\mathbf{A} = \begin{bmatrix} \mathbf{K} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{O} \end{bmatrix} \quad (4.6)$$

$$\Delta \mathbf{x} = \begin{Bmatrix} \Delta \mathbf{X} \\ \Delta \mathbf{Y} \end{Bmatrix} \quad (4.7)$$

$$\mathbf{B}^\star = \begin{bmatrix} \overline{\mathbf{B}} \\ \mathbf{O} \end{bmatrix}; \quad \mathbf{B}^{\star T} = \begin{bmatrix} \overline{\mathbf{B}}^T & \mathbf{O} \end{bmatrix} \quad (4.8)$$

permitindo a reescrita do sistema (4.5) para a seguinte forma compacta:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^\star \\ \mathbf{B}^{\star T} & \mathbf{O} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{x} \\ \overline{\mathbf{Y}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{O} \\ \bar{t}_\Gamma - \overline{\mathbf{B}}^T \mathbf{X} \end{Bmatrix} \quad (4.9)$$

O sistema (4.9) na forma explícita, vem

$$\mathbf{A} \Delta \mathbf{x} = -\mathbf{B}^{\star T} \overline{\mathbf{Y}} \quad (4.10)$$

$$\mathbf{B}^{\star T} \Delta \mathbf{x} = \bar{t}_\Gamma - \overline{\mathbf{B}}^T \mathbf{X} \quad (4.11)$$

Define-se agora o gradiente $\dot{\mathbf{x}}$ da variação da solução corrente através da seguinte equação:

$$\Delta \mathbf{x} = \dot{\mathbf{x}} \overline{\mathbf{Y}} \quad (4.12)$$

Substituindo a definição (4.12) na equação (4.10) obtém-se

$$\mathbf{A} \dot{\mathbf{x}} = -\mathbf{B}^\star \quad (4.13)$$

O sistema (4.13) pode agora ser resolvido para determinar o gradiente $\dot{\mathbf{x}}$ da solução. A matriz dos coeficientes \mathbf{A} é a mesma da iteração anterior, sendo guardada na sua forma decomposta para evitar um novo processo de factorização. O peso associado $\overline{\mathbf{Y}}$ ao novo grau de liberdade $\overline{\mathbf{Z}}$ pode agora ser determinado.

Para isso, é possível simplificar a segunda equação do sistema (4.9), substituindo a equação (4.12) em (4.11), tomando o aspecto,

$$\mathbf{B}^{\star T} \dot{\mathbf{x}} \bar{\mathbf{Y}} = \bar{t}_\Gamma - \bar{\mathbf{B}}^T \mathbf{X} \quad (4.14)$$

Resolvendo a equação anterior para $\bar{\mathbf{Y}}$,

$$\bar{\mathbf{Y}} = \frac{\bar{t}_\Gamma - \bar{\mathbf{B}}^T \mathbf{X}}{\mathbf{B}^{\star T} \dot{\mathbf{x}}} \quad (4.15)$$

tendo em conta as seguintes condições

$$\mathbf{B}^{\star T} \dot{\mathbf{x}} \neq 0 \quad (4.16)$$

e

$$\bar{t}_\Gamma - \bar{\mathbf{B}}^T \mathbf{X} \neq 0 \quad (4.17)$$

A primeira condição assegura a não existência de *modos espúrios* quando o denominador da equação (4.15) é igual a zero. Modos espúrios de energia nula aparecem quando alguns termos da base de aproximação de ordem mais alta que de outra forma iriam contribuir para a energia (de deformação no caso de problemas de elasticidade) não o fazem porque as suas contribuições anulam-se nos pontos de Gauss da integração, não tendo portanto, significado físico. A segunda condição (4.17) quando acopolada com o denominador nulo não tem solução, e indeterminado quando esta também é zero. É importante referir que uma variação nula de energia não significa automaticamente a existência de um modo espúrio, pois é matematicamente possível, ainda que improvável, a ocorrência de modos de energia nula sem ser espúrios, pois a matriz \mathbf{K} não é positivamente definida.

A partir do numerador da equação (4.15) é obtido o indicador do erro na fronteira ou norma da densidade do erro, ε_Γ , na fronteira, descrito na equação (4.18)

$$\varepsilon_\Gamma = \left\| \frac{\bar{t}_\Gamma - \Delta \bar{t}_\Gamma}{L} \right\| \quad (4.18)$$

sendo $\Delta \bar{t}_\Gamma = \bar{\mathbf{B}}^T \mathbf{X}$ e L o comprimento da fronteira.

A energia da iteração actual pode ser calculada somando a energia da iteração anterior com a variação de energia que a variação da solução da presente iteração provoca,

$$U_{i+1} = U_i + \Delta U_{i+1} \quad (4.19)$$

sendo

$$U_i = \frac{1}{2} (\mathbf{X}_i^T \mathbf{K}_i \mathbf{X}_i) \quad (4.20)$$

e

$$\Delta U_{i+1} = \frac{1}{2}(\Delta \mathbf{X}_{i+1}^T \mathbf{K}_i \Delta \mathbf{X}_{i+1}) \quad (4.21)$$

logo, a sua forma expandida será

$$U_{i+1} = \frac{1}{2}(\mathbf{X}_i + \Delta \mathbf{X})^T \mathbf{K}_i (\mathbf{X}_i + \Delta \mathbf{X}) \quad (4.22)$$

ou de forma ainda mais objectiva

$$U_{i+1} = \underbrace{\frac{1}{2}(\mathbf{X}_i^T \mathbf{K}_i \mathbf{X}_i)}_{\text{energia da iteração i}} + \underbrace{(\mathbf{X}_i^T \mathbf{K}_i \Delta \mathbf{X}_{i+1} + \frac{1}{2}(\Delta \mathbf{X}_{i+1}^T \mathbf{K}_i \Delta \mathbf{X}_{i+1}))}_{\text{variação de energia da iteração i+1}} \quad (4.23)$$

4.1.4 Refinamento no domínio

O refinamento no domínio, por sua vez, consiste em aumentar os graus de liberdade na base de aproximação do domínio, ou seja, acrescentar termos monomiais aos já existentes. A nova base de aproximação é expressa da seguinte forma:

$$\bar{\mathbf{T}} = \mathbf{U}(\mathbf{X} + \Delta \mathbf{X}) + \bar{\mathbf{U}} \bar{\mathbf{X}} \quad (4.24)$$

Assumindo que os graus de liberdade *completos* adicionados $\bar{\mathbf{X}}$ produzem uma variação, $\Delta \mathbf{X}$ e $\Delta \mathbf{Y}$, na solução, e adicionando um novo conjunto de equações de equilíbrio ao sistema resolvente, fica-se com o sistema de equações da solução refinada seguinte,

$$\begin{bmatrix} \mathbf{K} & \mathbf{B} & \bar{\mathbf{K}}_1 \\ \mathbf{B}^T & \mathbf{O} & \bar{\mathbf{B}}^T \\ \bar{\mathbf{K}}_1^T & \bar{\mathbf{B}} & \bar{\mathbf{K}} \end{bmatrix} \begin{Bmatrix} \mathbf{X} + \Delta \mathbf{X} \\ \mathbf{Y} + \Delta \mathbf{Y} \\ \bar{\mathbf{X}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{b} - \mathbf{q}_r \\ \mathbf{t}_r \\ -\bar{\mathbf{q}}_r \end{Bmatrix} \quad (4.25)$$

As contribuições dos novos graus de liberdade para a matriz de rigidez, as matrizes $\bar{\mathbf{K}}_1$ e $\bar{\mathbf{K}}$, são determinadas da equação (2.45). A matriz $\bar{\mathbf{K}}_1$ utiliza a base \mathbf{U} que contém todos os graus de liberdade até à presente iteração, e $\bar{\mathbf{U}}$ que contém apenas a base associada aos graus de liberdade adicionados, enquanto a matriz $\bar{\mathbf{K}}$ usa apenas a base com os novos graus de liberdade. A matriz de equilíbrio $\bar{\mathbf{B}}$ é determinada de (2.46) e também usa a base $\bar{\mathbf{U}}$ e a base na fronteira \mathbf{Z} até a presente iteração. Sendo o grau de liberdade estritamente dependente do elemento, todas as entradas do vector $\bar{\mathbf{K}}_1$ são nulas excepto as associadas com os graus de liberdade de \mathbf{X} atribuídas ao elemento enriquecido. De forma análoga, todas as entradas do

vector $\overline{\mathbf{B}}$ são nulas, excepto as associadas com os graus de liberdade de \mathbf{Y} atribuídos à fronteira do elemento enriquecido.

O sistema pode ser escrito como o conjunto de equações (4.26),

$$\begin{cases} \mathbf{K}\mathbf{X} + \mathbf{K}\Delta\mathbf{X} + \mathbf{B}\mathbf{Y} + \mathbf{B}\Delta\mathbf{Y} + \overline{\mathbf{K}}_1\overline{\mathbf{X}} = \mathbf{b} - \mathbf{q}_\Gamma \\ \mathbf{B}^T\mathbf{X} + \mathbf{B}^T\Delta\mathbf{X} + \overline{\mathbf{B}}^T\overline{\mathbf{X}} = \mathbf{t}_\Gamma \\ \overline{\mathbf{K}}_1^T\mathbf{X} + \overline{\mathbf{K}}_1^T\Delta\mathbf{X} + \overline{\mathbf{B}}\mathbf{Y} + \overline{\mathbf{B}}\Delta\mathbf{Y} + \overline{\mathbf{K}}\overline{\mathbf{X}} = -\overline{\mathbf{q}}_\Gamma. \end{cases} \quad (4.26)$$

Resolvendo apenas para a variação imposta à solução e tendo em conta que \mathbf{X} e \mathbf{Y} satisfazem a equação (2.58), resulta no seguinte conjunto de equações,

$$\begin{cases} \mathbf{K}\Delta\mathbf{X} + \mathbf{B}\Delta\mathbf{Y} + \overline{\mathbf{K}}_1\overline{\mathbf{X}} = \mathbf{0} \\ \mathbf{B}^T\Delta\mathbf{X} + \overline{\mathbf{B}}^T\overline{\mathbf{X}} = \mathbf{0} \\ \overline{\mathbf{K}}_1^T\Delta\mathbf{X} + \overline{\mathbf{B}}\Delta\mathbf{Y} + \overline{\mathbf{K}}\overline{\mathbf{X}} = -\overline{\mathbf{q}}_\Gamma - \overline{\mathbf{K}}_1^T\mathbf{X} - \overline{\mathbf{B}}\mathbf{Y} \end{cases} \quad (4.27)$$

que de forma compacta tem a seguinte estrutura:

$$\begin{bmatrix} \mathbf{K} & \mathbf{B} & \overline{\mathbf{K}}_1 \\ \mathbf{B}^T & \mathbf{O} & \overline{\mathbf{B}}^T \\ \overline{\mathbf{K}}_1^T & \overline{\mathbf{B}} & \overline{\mathbf{K}} \end{bmatrix} \begin{Bmatrix} \Delta\mathbf{X} \\ \Delta\mathbf{Y} \\ \overline{\mathbf{X}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{O} \\ \mathbf{O} \\ -\overline{\mathbf{q}}_\Gamma - \overline{\mathbf{K}}_1^T\mathbf{X} - \overline{\mathbf{B}}\mathbf{Y} \end{Bmatrix} \quad (4.28)$$

Para simplificar, as seguintes submatrizes podem ser definidas:

$$\mathbf{A} = \begin{bmatrix} \mathbf{K} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{O} \end{bmatrix} \quad (4.29)$$

$$\Delta\mathbf{x} = \begin{bmatrix} \Delta\mathbf{X} \\ \Delta\mathbf{Y} \end{bmatrix} \quad (4.30)$$

$$\overline{\mathbf{C}}^T = \begin{bmatrix} \overline{\mathbf{K}}_1^T & \overline{\mathbf{B}} \end{bmatrix} ; \overline{\mathbf{C}} = \begin{bmatrix} \overline{\mathbf{K}}_1 \\ \overline{\mathbf{B}}^T \end{bmatrix} \quad (4.31)$$

$$\Delta\overline{\mathbf{q}}_\Gamma = -\overline{\mathbf{K}}_1^T\mathbf{X} - \overline{\mathbf{B}}\mathbf{Y} \quad (4.32)$$

O sistema (4.28) pode então ser reestruturado da seguinte maneira,

$$\begin{bmatrix} \mathbf{A} & \overline{\mathbf{C}} \\ \overline{\mathbf{C}}^T & \overline{\mathbf{K}} \end{bmatrix} \begin{Bmatrix} \Delta\mathbf{x} \\ \overline{\mathbf{X}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{O} \\ -\overline{\mathbf{q}}_\Gamma + \Delta\overline{\mathbf{q}}_\Gamma \end{Bmatrix} \quad (4.33)$$

de onde é possível deduzir da primeira equação do sistema:

$$\mathbf{A}\Delta\mathbf{x} = -\overline{\mathbf{C}}\overline{\mathbf{X}} \quad (4.34)$$

Assumindo $\dot{\mathbf{x}}$ como gradiente da solução corrente,

$$\Delta\mathbf{x} = \dot{\mathbf{x}}\overline{\mathbf{X}} \quad (4.35)$$

a equação (4.34) é reescrita na seguinte forma:

$$\mathbf{A}\dot{\mathbf{x}} = -\overline{\mathbf{C}} \quad (4.36)$$

Da segunda equação do sistema (4.33) e tendo novamente presente a equação (4.35) obtém-se

$$\begin{aligned} \overline{\mathbf{C}}^T \Delta\mathbf{x} + \overline{\mathbf{K}}\overline{\mathbf{X}} &= -(\overline{\mathbf{q}}_{\Gamma} - \Delta\overline{\mathbf{q}}_{\Gamma}) \\ \overline{\mathbf{C}}^T \dot{\mathbf{x}}\overline{\mathbf{X}} + \overline{\mathbf{K}}\overline{\mathbf{X}} &= -(\overline{\mathbf{q}}_{\Gamma} - \Delta\overline{\mathbf{q}}_{\Gamma}) \\ \left[\overline{\mathbf{C}}^T \dot{\mathbf{x}} + \overline{\mathbf{K}} \right] \overline{\mathbf{X}} &= -(\overline{\mathbf{q}}_{\Gamma} - \Delta\overline{\mathbf{q}}_{\Gamma}) \end{aligned} \quad (4.37)$$

e introduzindo a nova variável $\dot{\mathbf{q}}$ como;

$$\dot{\mathbf{q}} = \overline{\mathbf{C}}^T \dot{\mathbf{x}} + \overline{\mathbf{K}} \quad (4.38)$$

resulta em,

$$\overline{\mathbf{X}} = -[\dot{\mathbf{q}}]^{-1} [\overline{\mathbf{q}}_{\Gamma} - \Delta\overline{\mathbf{q}}_{\Gamma}]. \quad (4.39)$$

É importante referir que caso $[\dot{\mathbf{q}}]$ seja singular ocorrem modos espúrios, e como tal esta deve ser analisada a cada iteração.

O indicador do erro no domínio ou norma da densidade do erro no domínio é definida como

$$\varepsilon_V = \left\| \frac{\overline{\mathbf{q}}_{\Gamma} - \Delta\overline{\mathbf{q}}_{\Gamma}}{V} \right\| \quad (4.40)$$

sendo V a área do domínio cujo erro é analisado.

Numa solução dita ‘óptima’, é desejável que seja distribuído equitativamente o erro da norma do domínio por todos os elementos, sendo desta forma, a solução mais equilibrada.

A nova energia é obtida somando a energia da iteração anterior com a variação de energia proporcionada, na iteração corrente, pelo refinamento no domínio, como a equação (4.41) indica

$$U_{i+1} = U_i + \Delta U_{i+1}. \quad (4.41)$$

Esta expressão pode ser reescrita na forma matricial na equação (4.42),

$$U_{i+1} = \mathbf{X}_{i+1}^T \mathbf{K}_{i+1} \mathbf{X}_{i+1} \quad (4.42)$$

ou na forma expandida, na equação (4.43)

$$U_{i+1} = \begin{pmatrix} \mathbf{X} + \Delta \mathbf{X} \\ \bar{\mathbf{X}} \end{pmatrix}^T \begin{bmatrix} \mathbf{K} & \bar{\mathbf{K}}_1 \\ \bar{\mathbf{K}}_1^T & \bar{\mathbf{K}} \end{bmatrix} \begin{pmatrix} \mathbf{X} + \Delta \mathbf{X} \\ \bar{\mathbf{X}} \end{pmatrix}. \quad (4.43)$$

4.1.5 Critérios de selecção de refinamento

O algoritmo p -adaptativo desenvolvido baseia-se no uso do critério de minimização da densidade do erro no domínio (ε_V) e a densidade do erro na fronteira (ε_Γ). Podem, no entanto, ser definidos critérios alternativos à minimização da densidade do erro, como o módulo da variação de energia ($\|\Delta U\|$) e a norma do incremento das temperaturas generalizadas ($\|\Delta \mathbf{X}\|$). A versão ‘completa’ do algoritmo refina todas as fronteiras e todos os domínios e, consoante a sua influência na variação da solução, seleccionar que fronteiras e domínios devem ser refinados. Esta versão pode ser reduzida para uma variante ‘parcial’ onde apenas o refinamento individual das fronteiras é efectuado com base num dos critérios escolhidos e a selecção de domínios a refinar depende de verificações subsequentes de indeterminação cinemática ou eliminação de ‘modos espúrios’. Foi esta variante escolhida para a aplicação desenvolvida. Esta opção deve-se ao facto que é mais simples de implementar e permite um maior controlo durante o processo de enriquecimento da base de aproximação nas fronteiras. A versão completa do algoritmo deverá ser desenvolvida e planeada como trabalho futuro.

O algoritmo permite que numa iteração se refina individualmente todas as fronteiras, ou de forma selectiva mais do que uma fronteira em simultâneo. Foi definido um parâmetro chamado ‘tolerância de selecção’ para controlar o nível de refinamento. Esta tolerância de selecção permite respeitar a simetria, caso exista, de um dado problema, visto que o refinamento de fronteiras simétricas relativas a um eixo, provocam a mesma variação na solução, e desta forma, são refinadas em simultâneo. A selecção individual de uma fronteira por iteração pode ser útil para problemas não simétricos, quando uma abordagem mais criteriosa é necessária, ainda que com mais iterações no processo iterativo.

O que se pretende com este parâmetro é garantir que, independentemente do critério escolhido, o seu valor é minimizado de forma relativamente uniforme em todas as fronteiras da malha, obter uma solução equilibrada e uma convergência

mais estável.

O processo de refinamento em cada iteração estima os valores ε_Γ , $\varepsilon_{\Delta U}$ e $\varepsilon_{\Delta X}$, para todas as fronteiras para um cenário de refinamento individual, sendo esta informação guardada na estrutura **Edges**. As fronteiras são ordenadas de acordo com o critério escolhido e todas que forem abrangidas pelo parâmetro da ‘tolerância de selecção’ são seleccionadas para refinamento. No algoritmo implementado, a ‘tolerância de selecção’ poderá ser 1%, 5% ou 15% de diferença do valor máximo do critério adoptado, ou nula, para problemas não-simétricos, então apenas a fronteira com o valor máximo listado é escolhida para refinamento.

O processo de refinamento no domínio, nesta versão do algoritmo, ocorre sempre após o refinamento na fronteira, e pode ou não acontecer de duas maneiras: o refinamento no domínio pode ocorrer caso existam modos espúrios detectáveis pelo refinamento na fronteira, o que impõe o refinamento dos elementos adjacentes à fronteira na qual foi identificado. A segunda forma de refinamento no domínio ocorrerá, após a verificação da indeterminação cinemática depois dos refinamentos prévios na presente iteração. Caso algum elemento da malha apresente um valor de indeterminação cinemática β menor a igual a zero, este elemento será aumentado um grau na base de funções de aproximação. Podem ocorrer refinamentos em múltiplos elementos na mesma iteração.

4.2 Descrição do algoritmo

Nesta secção é apresentada a estrutura do algoritmo de refinamento e o processo de tomada de decisão por parte do programa. O algoritmo é explicado de forma sucinta por meio de pseudo-código e, posteriormente, sobre a forma esquemática de *fluxograma*, na Figura 4.1.

1. Inserir dados: definição do problema, os graus iniciais das bases de aproximação, valores de tolerância estabelecidos, valores mínimos e máximos para diferentes parâmetros utilizados ao longo do programa.
2. Primeira verificação do programa; os graus iniciais escolhidos devem respeitar a condição de grau mínimo que o utilizador desejar, caso contrário todos os graus nos domínios e/ou fronteiras serão aumentados unitariamente.
3. De seguida, a indeterminação cinemática ($\beta \geq 0$) é verificada para cada elemento da malha e caso seja hipercinemática o programa segue para o passo seguinte, caso não seja, o grau do elemento é aumentado um grau de forma a alterar a indeterminação cinemática e o processo retorna ao passo (2).

4. Neste passo, é efectuada a análise do sistema com os parâmetros inicialmente propostos pelo utilizador. Os valores obtidos servem de referência e ponto de partida para o processo de refinamento adaptativo quando não se conhece à partida a solução exacta do problema.
5. Neste passo faz-se o refinamento na fronteira, obtém-se uma estimativa da norma da densidade do erro na fronteira, a variação de energia e a norma do incremento das temperaturas generalizada, para todas as fronteiras da malha refinada, uma a uma.
6. Todas as fronteiras e respectivas influências na solução devido ao refinamento são ordenadas e marcadas para refinamento com um dos critérios escolhidos (ε_Γ , ΔU ou $\Delta \mathbf{X}$). Os possíveis modos espúrios ($\varepsilon_\Gamma = 0$) são também identificados para posterior análise e refinamento.
7. A partir da lista criada do passo anterior, das fronteiras com possíveis modos espúrios, os elementos adjacentes são refinados, isto é, o grau da base de aproximação no domínio é aumentado em 1, ‘Refinamento no Domínio 1’ ou (DR1).
8. De novo a indeterminação cinemática de todos os elementos é verificada, e todos os elementos com $\beta \leq 0$ são refinados com o aumento de 1 grau na base de aproximação no domínio, ‘Refinamento no Domínio 2’ ou (DR2).
9. Em seguida, o número de condição do sistema governativo é verificado e caso ultrapasse um valor pré-definido, *threshold*, todos os elementos adjacentes às fronteiras refinadas na presente iteração são refinados.
10. Em cada iteração, os graus estabelecidos são verificados para ver se ultrapassam o valor máximo previamente estabelecido e prosseguir para o passo seguinte. Se os graus ultrapassarem o grau máximo o processo é interrompido, podendo haver a necessidade de intervenção do utilizador.
11. Para a conclusão do algoritmo, a convergência de energia é o critério de paragem. Se verificado, o programa segue para a análise final com conjunto de bases de aproximação actuais. Não se verificando, o processo iterativo é recommçado a partir do ponto (5).

Valores Limites ou de Tolerância

Durante o processo iterativo, alguns valores são definidos como valores limites ou de tolerância, uns pelo utilizador, outros programados directamente no código. Es-

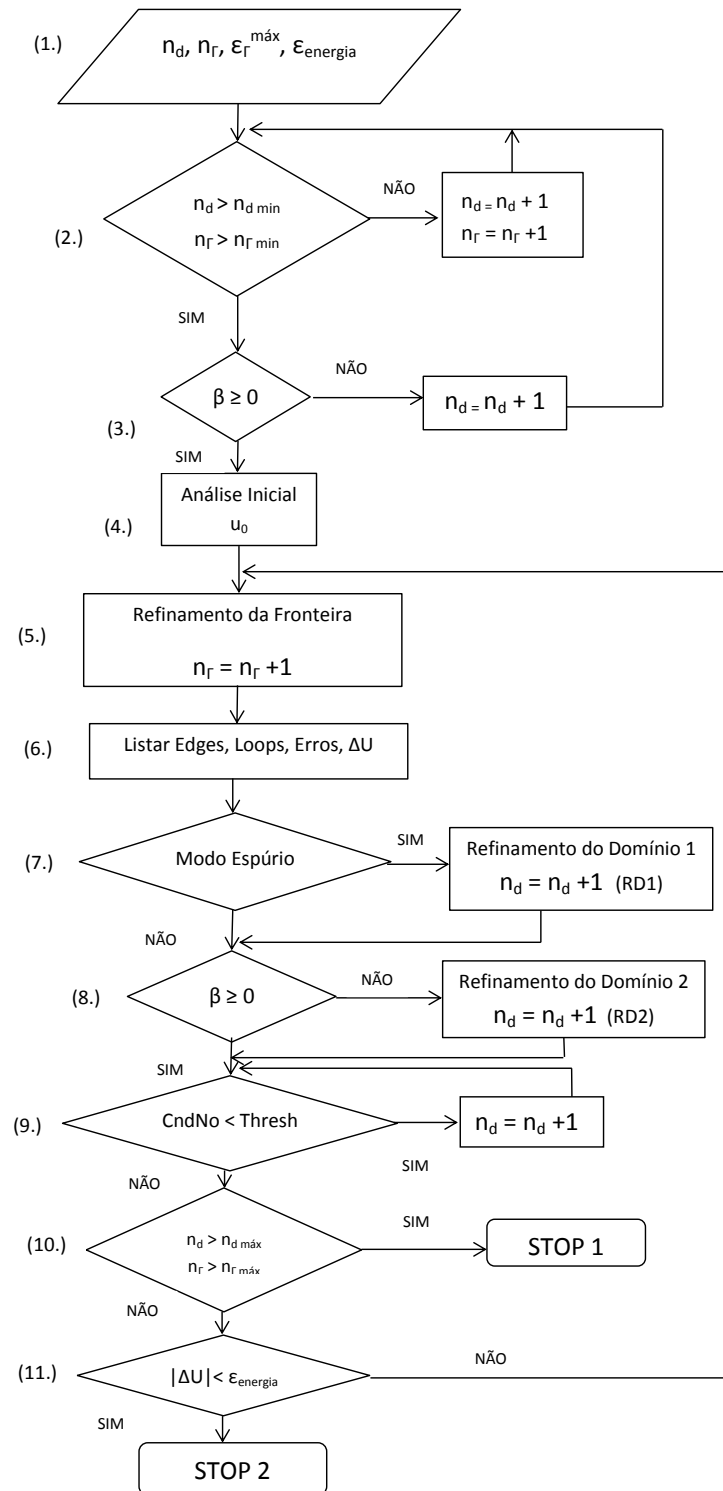


Figura 4.1: Fluxograma do programa desenvolvido

ses valores são usados no algoritmo automatizado para tomar decisões, durante o processo de refinamento.

Para controlar a convergência da energia é calculado o erro relativo da variação da energia ε_U , de uma das seguintes formas: em relação à iteração anterior ou o valor médio de um intervalo de iterações, de maneira a evitar extremos locais na evolução da energia. Nesta versão, o algoritmo está programado para as últimas três iterações. A convergência considera-se atingida se ε_U for menor a um valor estipulado. Na presente implementação do trabalho foram fixados os valores 10^{-4} ou 10^{-8} . Os dois limites foram estabelecidos para testar o desempenho do algoritmo.

De forma a controlar a degradação do sistema do ponto de vista numérico durante o refinamento, o número de condição é verificado para um limite pré-definido chamado *threshold*, com o valor de 10^{-5} . Quando este é atingido, são refinados os domínios adjacentes às fronteiras refinadas nessa iteração, mantendo o condicionamento do problema aceitável e com soluções viáveis.

Durante o processo iterativo são realizadas várias verificações aos graus das bases de aproximação. A verificação aos graus mínimos é feita apenas uma vez, no início do programa, e verifica se para a análise inicial, os graus determinados para as bases nas fronteiras e nos domínios introduzidos pelo utilizador não provocam indeterminação cinemática em qualquer dos elementos da malha. Caso ocorra, todos os elementos serão refinados, a malha re-testada para $\beta \leq 0$ e o processo repetido até a condição ser garantida. Os graus máximos são verificados no fim de cada iteração e são definidos no código, e, na presente versão do algoritmo são 15 para os domínios e 30 para as fronteiras. Caso algum dos graus seja atingido para quaisquer dos domínios ou fronteiras, o conjunto de graus actual é adoptado como o final.

4.3 Aspectos de implementação

O algoritmo é implementado de forma a ser inserido na fase de pré-processamento de um programa existente, estando dividido em diversos módulos, sendo facilmente adaptável ou acrescentado, caso assim se deseje.

É possível, devido a esse aspecto, uma leitura e compreensão simplificada do código, providenciando uma melhor manutenção e desempenho optimizado por meio da programação estruturada. A organização modular tem como finalidade primária auxiliar na gestão de sistemas (ou programas) que, de outra forma, seriam significativamente mais complexos.

Cada módulo pode conter vários processos, independentes de outros módulos, respeitando uma hierarquia. Nesta hierarquia, várias camadas de módulos podem ser invocadas, até finalmente atingir o módulo da camada principal. Este tipo de estruturas de dados permitem a hierarquização da informação e do código em si.

Do ponto de vista algorítmico, é realizada uma análise inicial, com os graus inicialmente escolhidos pelo utilizador, de forma a ter um primeiro termo de comparação para a evolução da energia no processo de refinamento. Posteriormente inicia-se o processo iterativo de refinamento, onde o programa irá continuamente refinar, de acordo com critérios previamente definidos, os graus das bases de funções de aproximação dos domínios e fronteiras, até se atingir convergência. Finalmente a partir dos graus definidos do algoritmo, a análise final é executada obtendo os campos de temperaturas e fluxos, e o processo de refinamento escrutinado na forma de tabelas e gráficos.

A formulação híbrida tem uma organização estrutural particular que permite manipulação do sistema de equações em blocos. Estes podem ser adicionados ou removidos ao sistema global durante o processo iterativo caso o refinamento adaptativo assim o requeira.

Durante o processo iterativo o sistema global é continuamente alterado, fruto do refinamento realizado, mas no fim de cada iteração, este é de novo reorganizado assimilando a variação imposta ao sistema que é, na prática, os graus de liberdade adicionados. Isto permite que seja acedido no início de cada iteração, sem necessidade de ser reconstruído, de forma a ser continuamente actualizado, salvaguardando o processo iterativo.

Capítulo 5

Resultados Numéricos

O algoritmo de refinamento descrito no capítulo anterior é aplicado aqui a três casos de estudo, previamente resolvidos em [41]. As razões para a escolha destes casos de estudo são explicadas e os resultados obtidos analisados e comparados com os esperados à partida. O algoritmo de refinamento é avaliado para diferentes critérios de selecção das fronteiras a refinar e tolerâncias de saída.

5.1 Caso de estudo 1 - problema bisimétrico

O primeiro caso de estudo onde o algoritmo é testado é um problema relativamente simples com a particularidade de ter dois eixos de simetria. A baixa exigência numérica deste problema permite averiguar o funcionamento básico do algoritmo e compreender o impacto de algumas das opções de controlo no seu desempenho numérico.

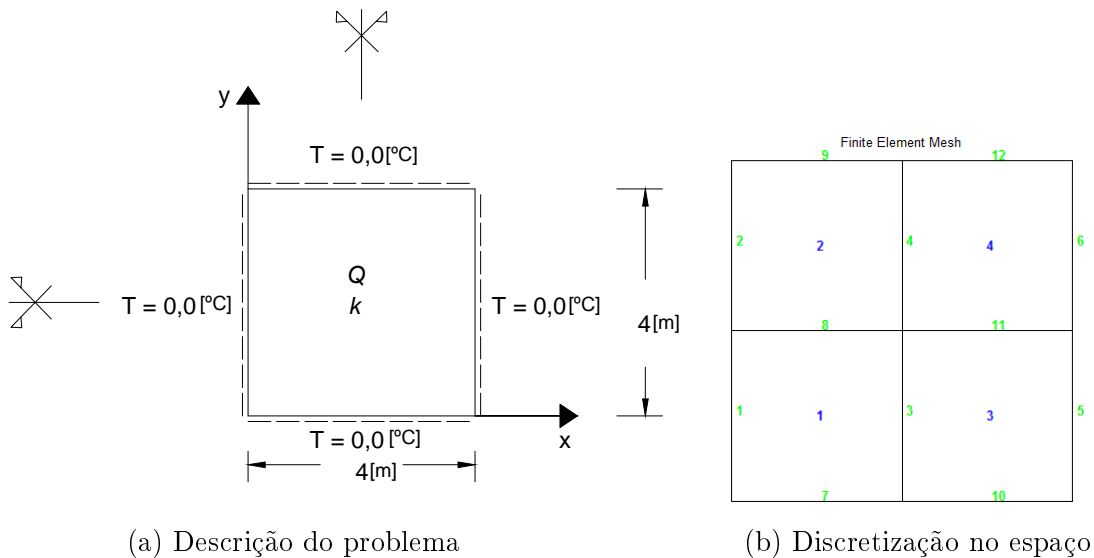


Figura 5.1: Caso de estudo 1 e malha utilizada na sua discretização

O problema é uma placa quadrada, representada na Figura 5.1a, com condutividade térmica (k) de $1 \text{ Wm}^{-1}\text{K}^{-1}$ e calor gerado internamente (Q) de 20 Wm^{-2} . Apenas existem condições de fronteira Dirichlet, com temperaturas impostas (T) de 0 °C para todas as fronteiras exteriores. O problema foi discretizado numa malha de quatro elementos, representada na Figura 5.1b, de dimensões iguais entre si. As bases de aproximação iniciais admitidas foram de grau 1 para as fronteiras e grau 3 para os domínios.

Para ilustrar a exigência numérica do problema em análise, apresentam-se primeiro os resultados obtidos em [41], para o mesmo problema e uma malha semelhante, e que servirão como termo de referência, para avaliação do algoritmo de refinamento. Assim, o campo de temperaturas obtido por Silva [41] encontra-se representado na Figura 5.2a, os graus utilizados na análise na Figura 5.2b e os campos de fluxos nas direcções x e y na Figura 5.3.

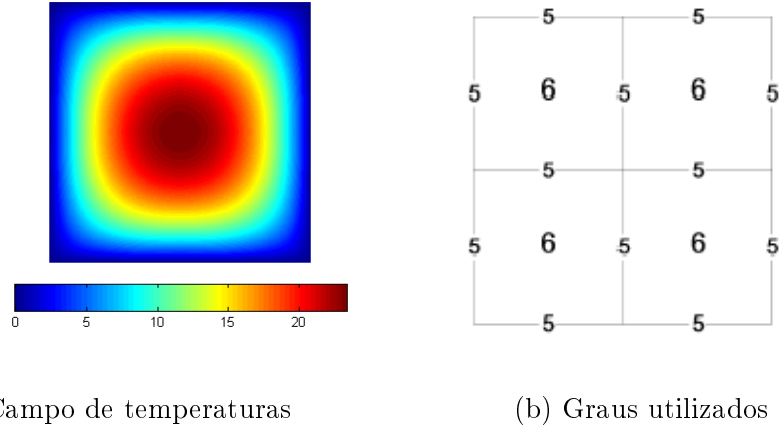


Figura 5.2: Campo de temperaturas e graus das bases de aproximação adoptados em [41]

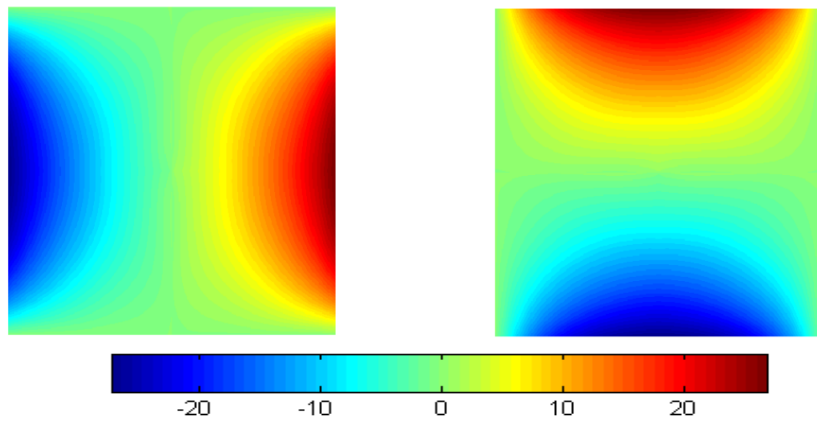


Figura 5.3: Campos de fluxos obtidos em [41] (esq q_x ; dir q_y)

O desempenho do algoritmo p -adaptativo será analisado de forma aprofundada apenas para um dos critérios de escolha dos refinamentos a aplicar. Para esse efeito,

foi escolhido o critério da norma da densidade do erro na fronteira. A comparação entre os três critérios propostos, definidos na Secção 4.1.5, será feita posteriormente. Devido à simetria do problema, foi utilizada uma tolerância de selecção baixa (1%) e uma tolerância de saída de 10^{-8} .

O processo iterativo do refinamento pode ser analisado de forma mais detalhada na Tabela C.1, presente no Apêndice C. Contudo, será estudada graficamente a evolução de vários parâmetros, como a energia da solução e respectiva convergência, na Figura 5.4; o módulo do erro da energia, definido como sendo a variação desta em iterações sucessivas, na Figura 5.5; e a norma da densidade do erro da(s) fronteira(s) seleccionada(s) para refinar, na Figura 5.6.

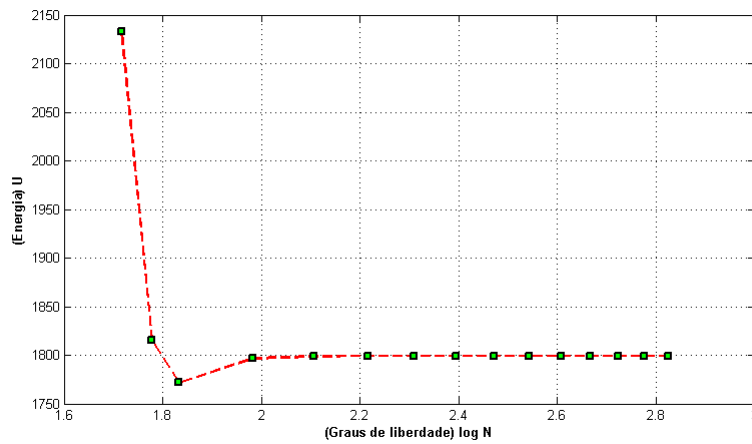


Figura 5.4: Variação da energia térmica no processo iterativo

O gráfico da Figura 5.4 tem representado, no eixo das ordenadas a energia, e no eixo das abscissas o número total de graus de liberdade, em escala logarítmica. Existe uma convergência nítida mas não monótona, como era expectável, e que já tinha sido previamente referida na Secção 4.1.2 como característica da formulação híbrida.

A iteração inicial está dependente do conjunto de graus iniciais dados às bases de aproximação introduzidas pelo utilizador, logo, é natural que esta apresente a energia mais afastada do valor convergido. Neste caso, o valor para o qual a energia convergiu foi de $1.7994e + 03$ que, quando comparado com o valor obtido em [41] de $1.7990e + 03$, aceita-se como adequado.

A norma do erro da energia regista a variação de energia em iterações consecutivas, e é, na prática, o mecanismo de saída do algoritmo de refinamento. No eixo das ordenadas está representado o erro da energia, e no eixo das abscissas o número de graus de liberdade, ambos em escala logarítmica. Quando o valor médio das últimas três iterações seja tal que cumpra a tolerância de saída definida, o refinamento é concluído e a convergência da energia assumida.

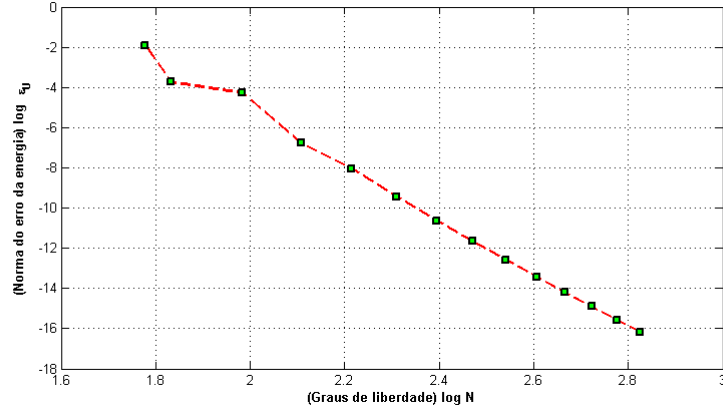


Figura 5.5: Norma do erro da energia no processo iterativo

Durante o refinamento, as variações de energia vão-se reduzindo progressivamente durante o processo iterativo de refinamento de uma forma relativamente constante. Neste caso, o refinamento cessa na 14ª iteração, quando o valor médio das três últimas iterações anteriores atinge a tolerância estabelecida.

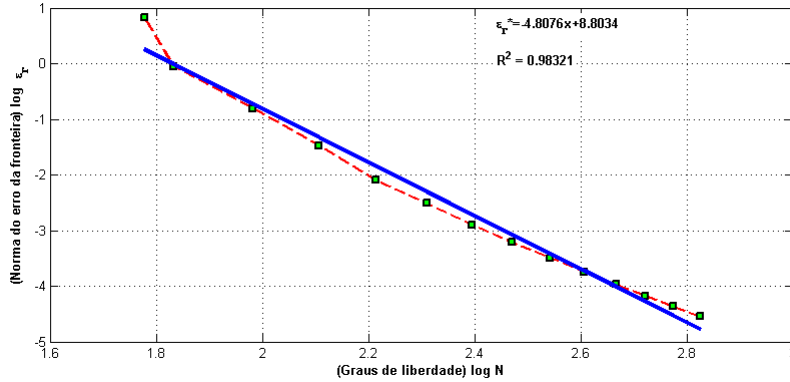


Figura 5.6: Norma da densidade do erro na fronteira no processo iterativo

O gráfico da norma da densidade do erro na fronteira, representado na Figura 5.6 tem, no eixo das abscissas, o número total de graus de liberdade e, no eixo das ordenadas, a norma da densidade do erro na fronteira, os dois em escala logarítmica. O gráfico demonstra uma clara tendência de diminuição da densidade do erro na fronteira durante o refinamento. Uma regressão linear do conjunto de dados é representada no gráfico a azul, com a expressão ϵ_f^* e o coeficiente de determinação R^2 que descreve o quão bem se ajusta a recta à amostra, com o valor 0 como o pior e 1 como o melhor. O valor, para este caso, foi de 0.983, comprovando a ideia que o comportamento é no sentido de ir diminuindo durante o processo iterativo.

A densidade do erro apresenta uma tendência estável decrescente na sua evolução, em iterações sucessivas. Tal verifica-se apesar da natureza da base de aproximação na fronteira, que contém, de forma alternada, funções pares e ímpares (característica

dos polinómios de Chebyshev) e que poderá causar alguma instabilidade.

Os graus das bases de aproximação no final do processo iterativo, com os graus nas fronteiras a verde claro, e os graus nos domínios a azul escuro são os apresentados na Figura 5.7.

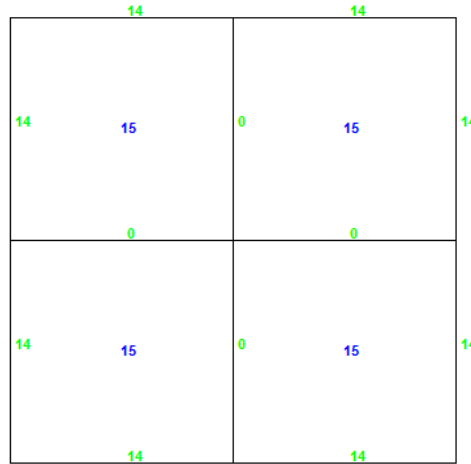


Figura 5.7: Graus finais das bases de aproximação

Como seria desejável, as bases de aproximação adoptadas pelo algoritmo mantêm a simetria em ambos os eixos. Caso a tolerância de selecção fosse nula, refinando assim apenas uma fronteira por iteração, tal poderia não ser garantido.

As fronteiras directamente sobre os eixos de simetria nunca são refinadas, pois nunca são seleccionadas pelo programa, visto que o seu refinamento não afeta a continuidade interelementar das temperaturas em virtude da sua simetria. Em contrapartida, as fronteiras exteriores são sempre seleccionadas, todas em conjunto, para refinamento em cada iteração. É o que se esperava porque, novamente devido à bi-simetria, o refinamento em qualquer uma destas fronteiras produz o mesmo efeito no critério escolhido. Logo, na listagem realizada para determinar quais fronteiras são refinadas, todas as fronteiras, dentro da tolerância admitida, são seleccionadas.

De seguida são apresentados os campos de temperatura e fluxos de calor nas direcções x e y , obtidos em cada passo do processo iterativo.

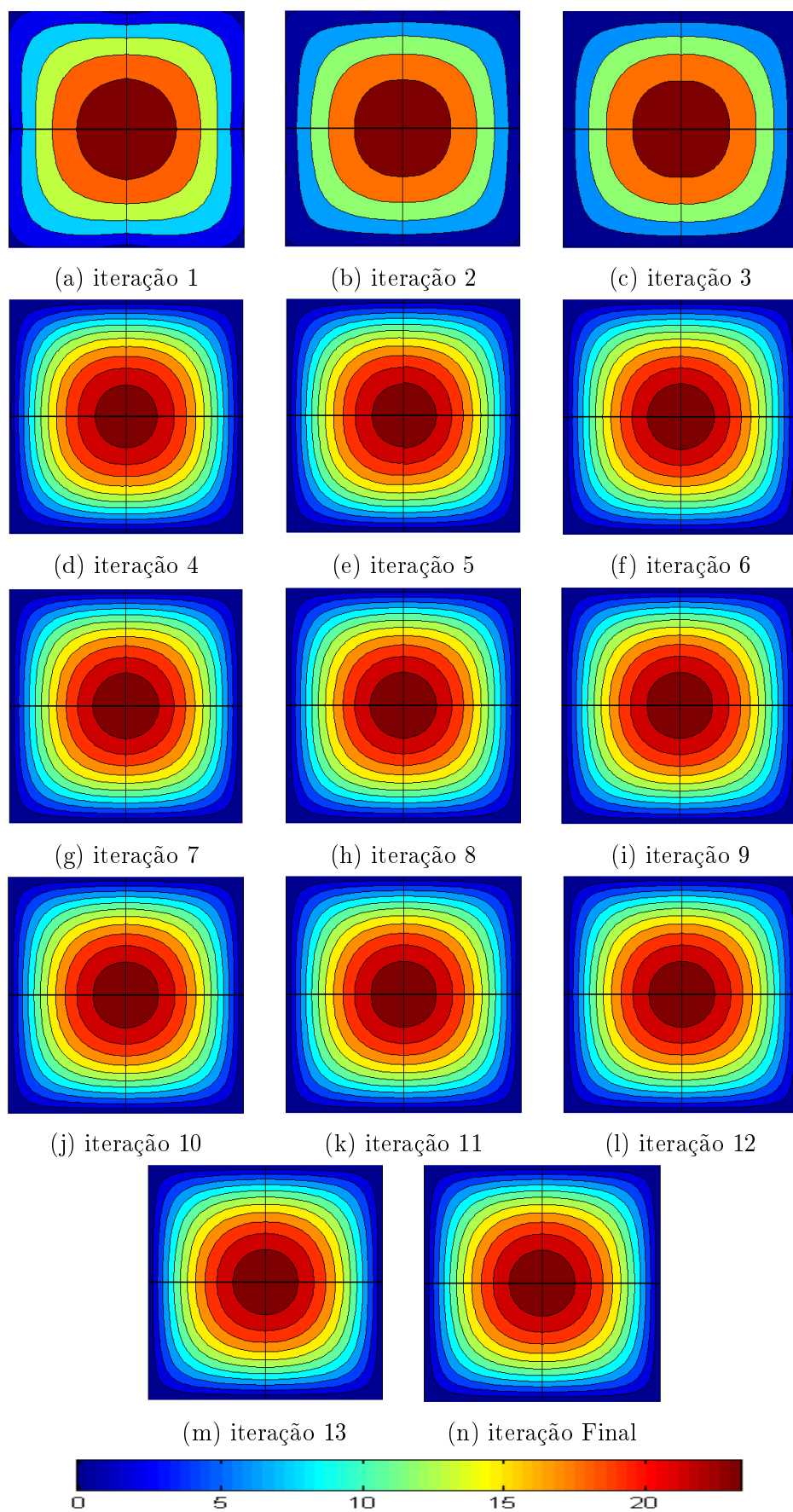
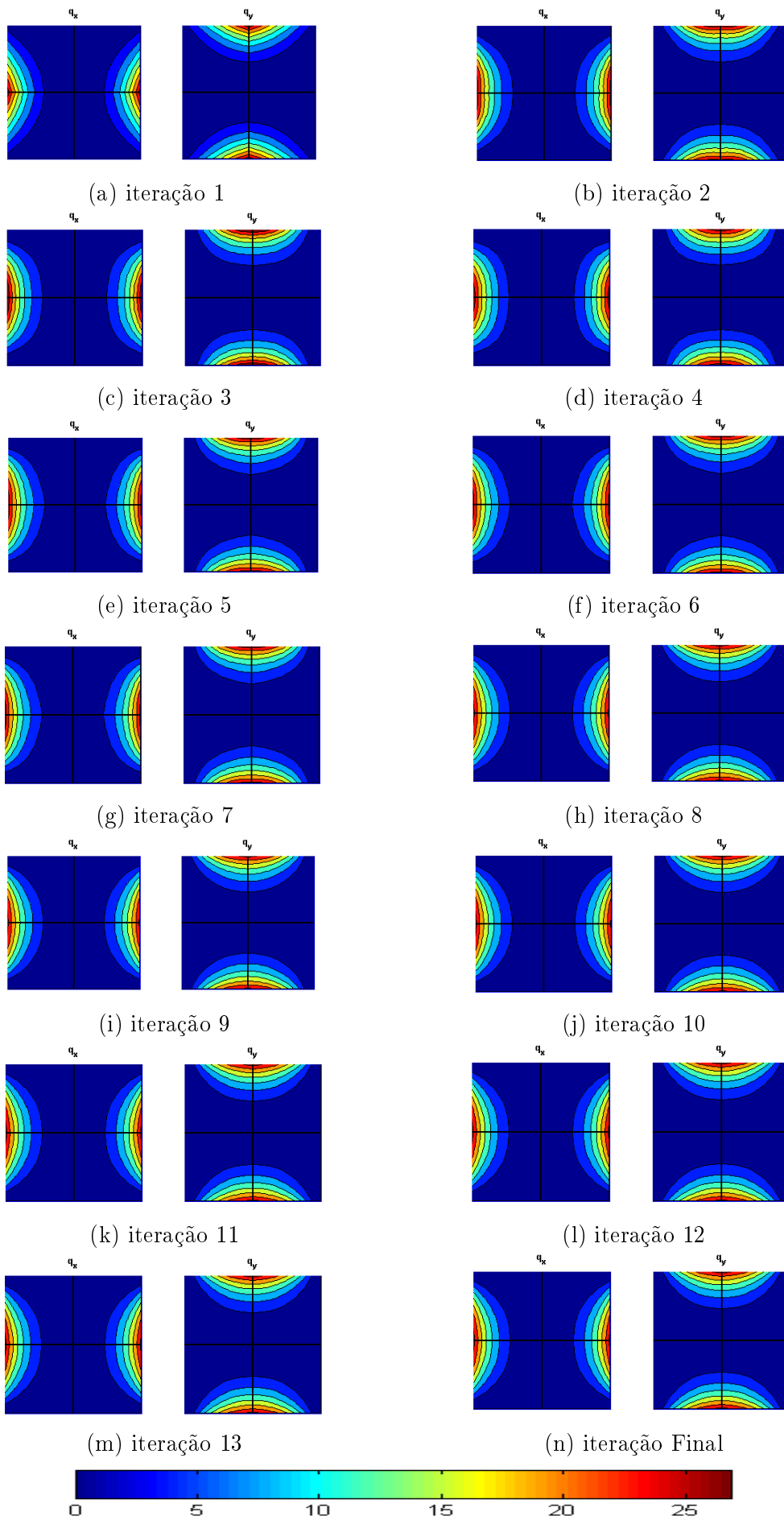


Figura 5.8: Campos de temperaturas

Figura 5.9: Campos de fluxos (esq q_x , dir q_y)

Os campos de temperaturas e fluxos apresentam um acréscimo de qualidade ao longo do refinamento, principalmente nas primeiras iterações. O aumento de graus de liberdade fornece informação cada vez mais precisa sobre a solução pelo que a sua representação se vai aperfeiçoando. No entanto, nas iterações finais, a melhoria é marginal, e visualmente imperceptível, consequência da convergência da solução. É importante referir que, nesta análise, os resultados nos campos de fluxos apresentam valores positivos para a normal exterior e negativos para a normal interior das fronteiras. Nos resultados obtidos em [41], os valores dos fluxos são dependentes dos sentidos que estes tomam em referência ao eixo global definido.

Comparação entre critérios

Para o caso de estudo 1, com dois eixos de simetria, não se verificaram diferenças entre os três critérios propostos para a selecção de fronteiras a refinar. Na Figura 5.10 estão representados a variação da energia térmica, a norma do seu erro e a norma da densidade do erro na fronteira, para cada critério. O critério da densidade do erro na fronteira está representado graficamente a vermelho, o critério da variação de energia a azul e o critério da norma do incremento das temperaturas generalizadas a magenta.

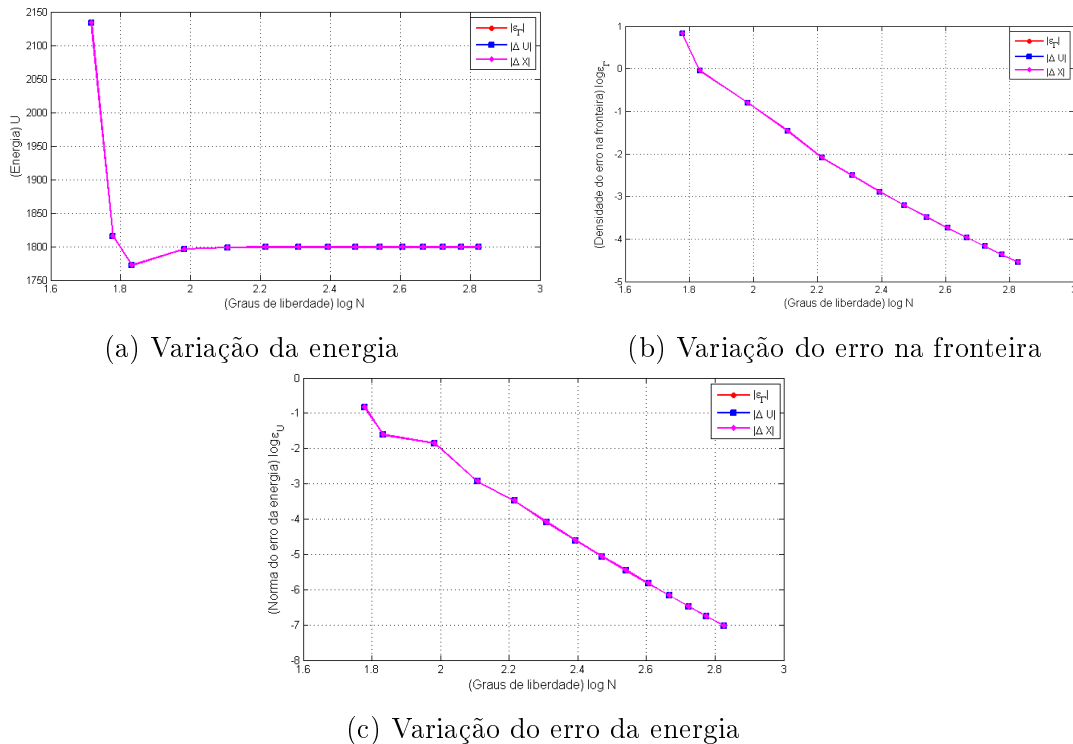


Figura 5.10: Comparação dos diferentes critérios com tolerância de selecção de 1%

Pode-se então concluir que todas as análises se sobrepõem, comprovando que não existe diferença entre os critérios, para o caso bisimétrico.

5.2 Caso de estudo 2 - problema assimétrico

O caso de estudo 2, representado na Figura 5.11a, é uma placa quadrada com condutividade térmica (k) de $30 \text{ W m}^{-1} \text{ K}^{-1}$ e geração interna de calor (Q) de 100 W m^{-2} . Ao contrário dos outros casos de estudo, este problema não apresenta qualquer eixo de simetria e tem em simultâneo condições de fronteira Dirichlet e Neumann, com temperaturas impostas de 50 °C nas fronteiras superior e direita e fluxos impostos (q) de 25 W m^{-2} na fronteira esquerda e 0 W m^{-2} na fronteira inferior. O problema foi discretizado numa malha de quatro elementos semelhante ao primeiro caso de estudo, representada na Figura 5.11b. As bases de aproximação iniciais admitidas foram de grau 1 para as fronteiras e de grau 3 para os domínios.

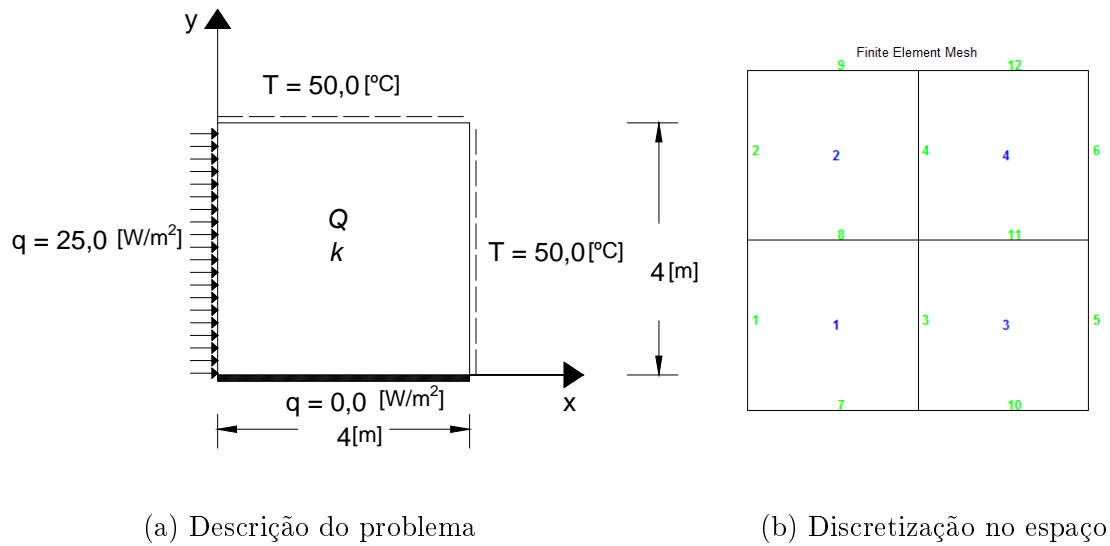


Figura 5.11: Caso de estudo 2 e a malha utilizada na sua discretização

Este problema foi resolvido na referência [41] utilizando uma malha semelhante e a mesma tipologia de elementos. Os campos de temperaturas e fluxos obtidos em [41], nas direcções x e y , são representados nas Figuras 5.12 e 5.13, respectivamente. Para estes resultados, os graus das bases de aproximação utilizados não foram fornecidos.

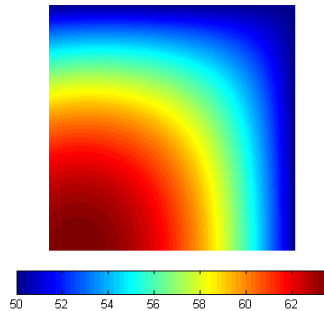


Figura 5.12: Campo de temperaturas em [41]

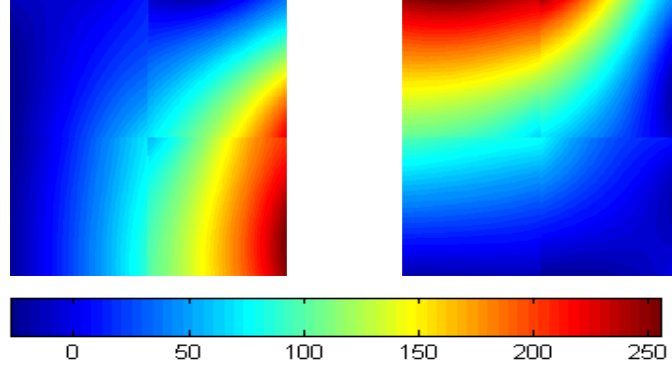


Figura 5.13: Campos de fluxos em [41] (esq q_x ; dir q_y)

O desempenho do algoritmo p -adaptativo será analisado de forma aprofundada apenas para o critério de selecção da densidade do erro na fronteira. A comparação dos três critérios de selecção descritos na Secção 4.1.5 será reservada para posterior estudo. Tendo em conta a falta de simetria da estrutura, é refinada apenas uma fronteira por iteração (a que obtiver maior erro). A tolerância de saída é de 10^{-8} .

As variações, durante o processo de refinamento, da energia da solução, na Figura 5.14, da norma do erro da energia na Figura 5.15 e da norma da densidade do erro na fronteira refinada na Figura 5.16, são apresentadas graficamente de seguida.

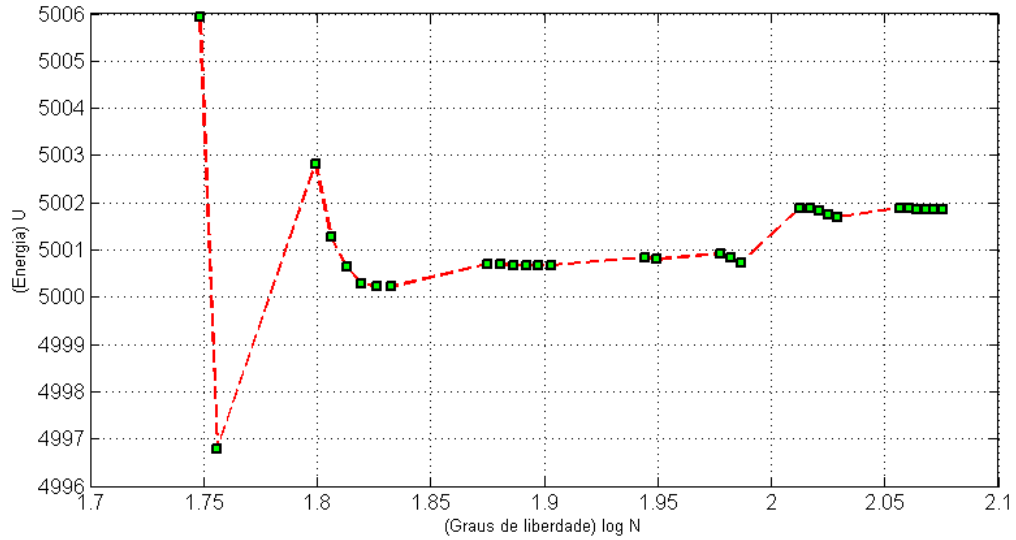


Figura 5.14: Variação da energia com os graus de liberdade

A evolução da energia, no eixo das ordenadas, registada durante o processo de refinamento, com o número total de graus de liberdade no eixo das abscissas em escala logarítmica, é similar com a do caso de estudo anterior. Apresenta inicialmente uma instabilidade devido ao fraco nível de refinamento inicial definido pelo utilizador, que se reduz durante o processo iterativo, convergindo para o valor de $5.0019e + 03$. O valor calculado em [41] foi de $5.0059e + 03$, sendo o resultado obtido adequado.

As oscilações seguintes, menores comparativamente, ocorrem quando existe refinamento no domínio cuja influência no sistema é maior do que o refinamento de uma fronteira.

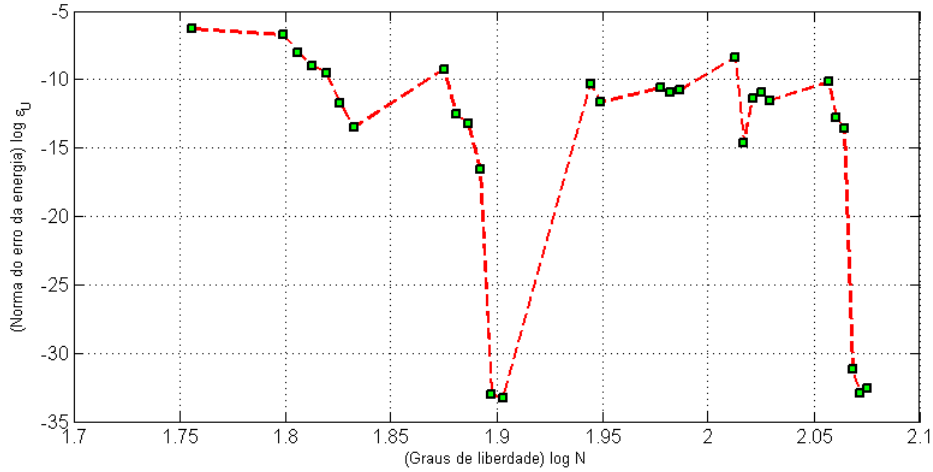


Figura 5.15: Norma do erro da energia no processo iterativo

O gráfico da Figura 5.15, apresenta a variação da norma da energia, com o número total de graus de liberdade. Ambos os eixos são logarítmicos. A norma do erro da energia, que mede o módulo da variação de energia em iterações consecutivas, tem também um comportamento semelhante ao primeiro caso de estudo. Os erros de energia têm tendência a diminuir ao longo do refinamento, excepto nas iterações em que ocorram refinamentos nos domínios, que, como acima foi mencionado, têm maior influência na energia do sistema. O refinamento cessou na 29ª iteração, quando o valor médio das últimas três iterações cumprir a tolerância de saída admitida.

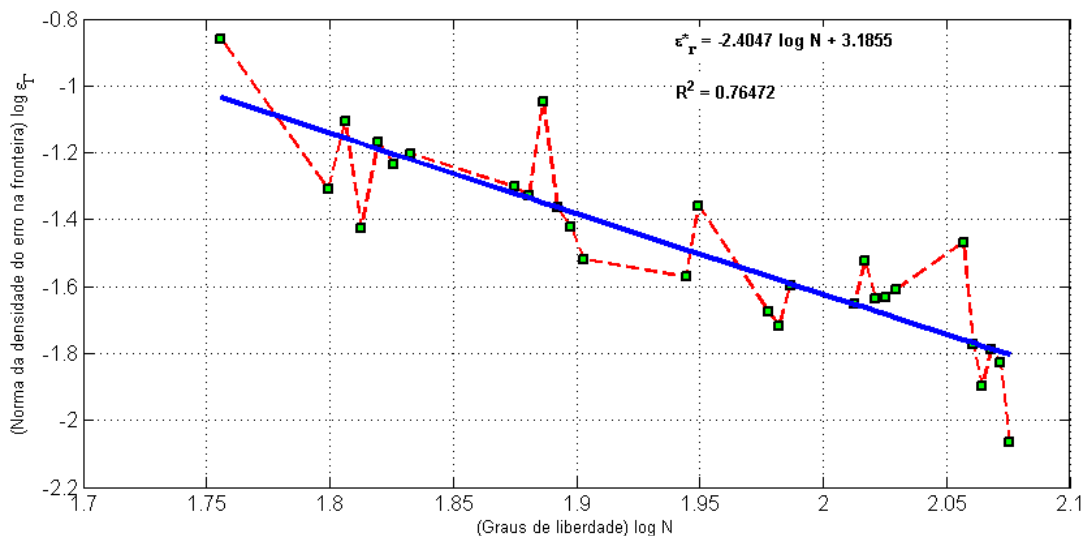


Figura 5.16: Norma da densidade do erro na fronteira no processo iterativo

O gráfico da Figura 5.16 tem, no eixo das abscissas, o número de graus de liberdade em escala logarítmica e, no eixo das ordenadas, a norma da densidade do erro na fronteira seleccionada para refinar, também em escala logarítmica. A norma do erro na fronteira tem tendência de diminuir ao longo do processo iterativo ainda que apresente a mesma instabilidade verificada no caso de estudo 1. Uma regressão linear dos erros foi adicionada ao gráfico a azul. O coeficiente de determinação R^2 tem o valor de 0.765, evidenciando a tendência global decrescente ao longo do processo de refinamento.

Os graus das bases de aproximação, no fim do processo de refinamento, tabelado na Tabela C.2 em anexo no Apêndice C, são esquematizados na Figura 5.17

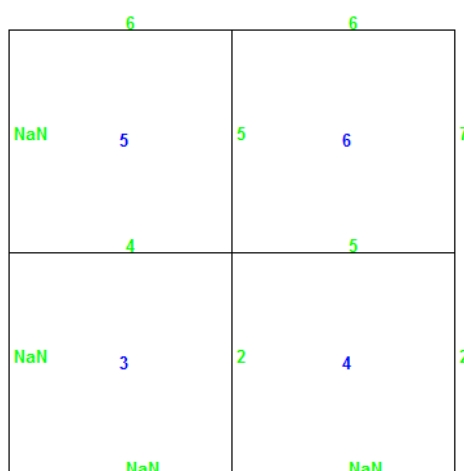


Figura 5.17: Graus finais das bases de aproximação

Como é possível observar, não se detecta qualquer simetria nos graus determinados pelo algoritmo, o que era previsível tendo em conta que este problema não possui qualquer eixo de simetria. É perceptível a tendência que o programa exhibe em seleccionar para refinamento com maior frequência, as fronteiras 9, 12 e 6 e os elementos 2, 3 e 4, por estarem mais próximos das fronteiras com temperaturas impostas onde se concentrarão os maiores gradientes de fluxos.

Nas páginas seguintes são apresentados os campos de temperatura e campos de fluxos de calor nas direcções x e y , obtidos do algoritmo desenvolvido. São apenas demonstrados os campos para as iterações pares do processo de refinamento, devido ao elevado número de iterações, de forma a poder acompanhar graficamente a evolução da solução.

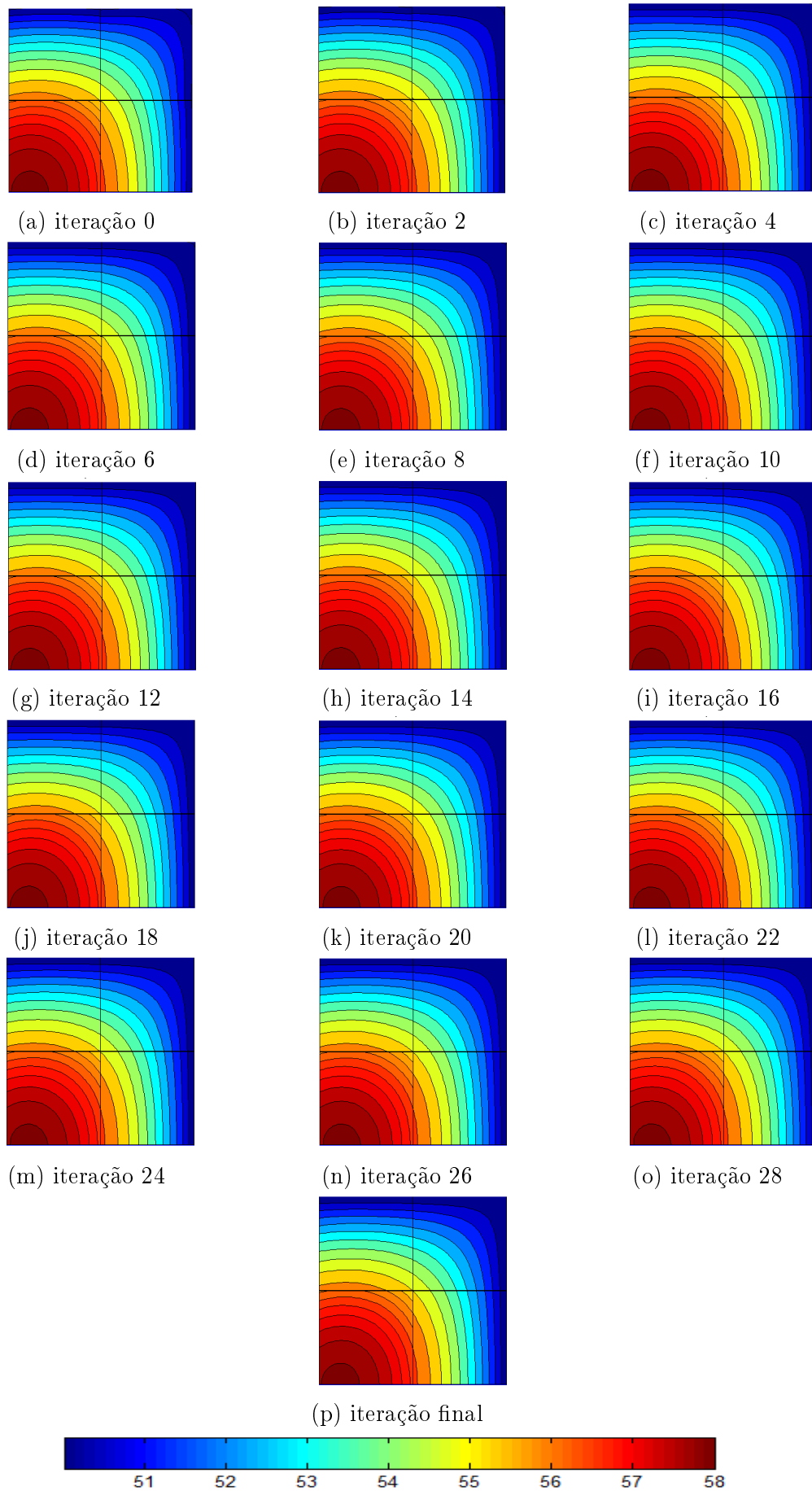


Figura 5.18: Campos de temperaturas

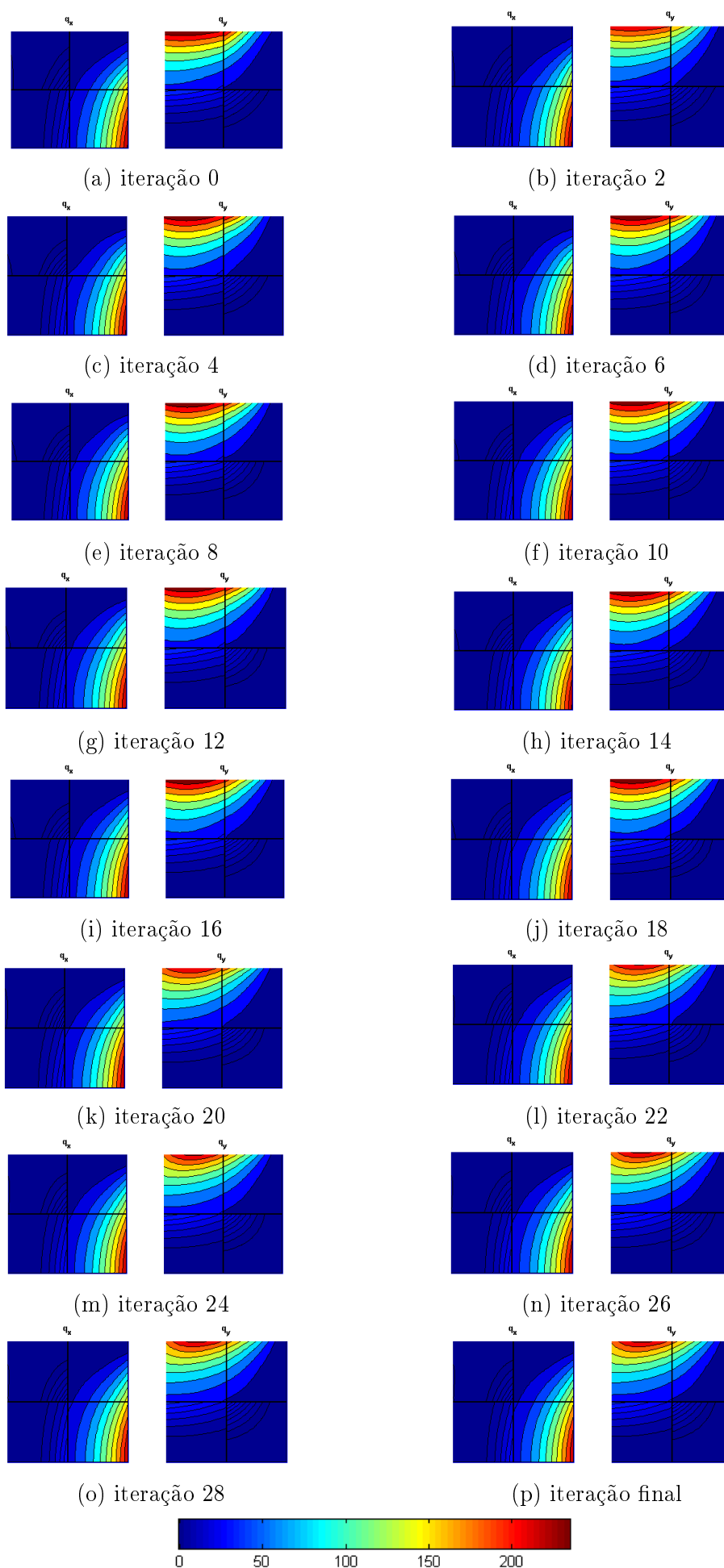


Figura 5.19: Campos de fluxos (esq q_x , dir q_y)

Os campos de temperaturas apresentam uma evolução muito pouco perceptível ao olho nu, com ligeiras melhorias nas limitações gráficas de cada campo de temperaturas. Os campos de fluxos já têm um desenvolvimento mais acentuado, nomeadamente a continuidade interelementar a ser melhorada com o aumento de iterações.

Comparação entre critérios

O refinamento é realizado escolhendo um de três critérios, referidos na Secção 4.1.5, no processo de selecção de fronteiras para refinar. Os critérios que vão ser analisados são a norma da densidade do erro na fronteira ε_Γ , o módulo da variação de energia $\|\Delta U\|$ ou a norma do incremento das temperaturas generalizadas $\|\Delta \mathbf{X}\|$. Cada critério exhibe um modo de actuar diferente dos outros.

A análise comparativa dos critérios será efectuada para três valores de tolerância de selecção das fronteiras, nomeadamente 0%, 5% e 15%. Enquanto que em problemas simétricos escolher uma tolerância de selecção não nula é desejável para não pôr em causa a simetria da solução, considerou-se interessante averiguar se definir uma tolerância relativamente alta em problemas não simétricos podia contribuir para a diminuição do esforço computacional através da redução do número de iterações necessárias para alcançar uma determinada precisão. No entanto, como vai ser demonstrado, o refinamento múltiplo de fronteiras pode ter, para alguns casos, o efeito contrário ao de uma convergência mais expedita.

Vai ser averiguado o comportamento do algoritmo para os três critérios de selecção diferentes utilizando, para cada um, três tolerâncias de selecção diferentes. Para além do objectivo de compreender o comportamento de cada critério, existe o objectivo de averiguar em que medida a utilização de uma tolerância de selecção mais alta torna o processo iterativo mais célere.

Na Figura 5.20 é apresentado a comparação de vários parâmetros com diferentes critérios para o caso de estudo 2. O critério da norma da densidade do erro na fronteira está representado graficamente a vermelho, o critério da variação da energia a azul e o critério da norma do incremento das temperaturas generalizadas a magenta. Neste caso, apenas uma fronteira foi seleccionada para refinamento por iteração (tolerância de selecção nula). Observa-se que, de acordo com o critério adoptado, a convergência ocorre a diferentes taxas, com o critério $\|\Delta \mathbf{X}\|$ a necessitar de menos iterações até atingir a convergência que o critério ε_Γ . O critério $\|\Delta U\|$, como se pode constatar, demonstra um comportamento intermédio entre os outros dois.

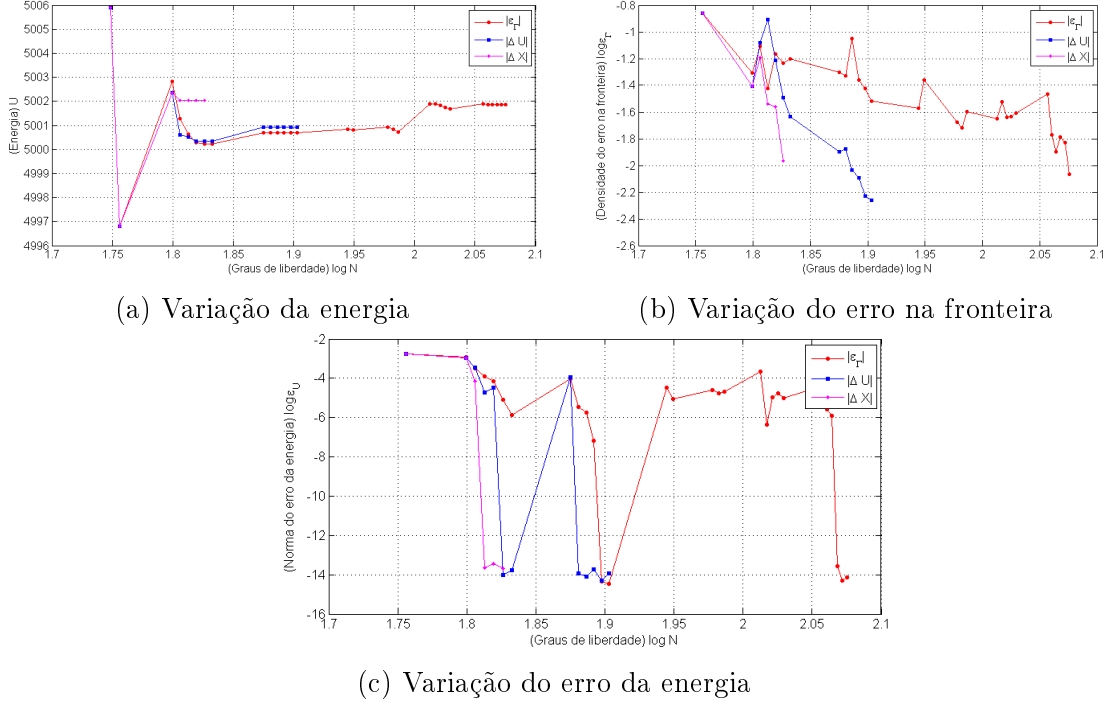


Figura 5.20: Comparação dos três critérios com tolerância nula de selecção

Na Figura 5.21 é apresentado de novo o mesmo caso de estudo, desta vez com tolerância de selecção de 5%, ou seja, com o algoritmo a seleccionar frequentemente duas fronteiras e esporadicamente três para refinamento. No entanto, o número de iterações necessárias é maior que para uma selecção de apenas uma fronteira refinada por iteração.

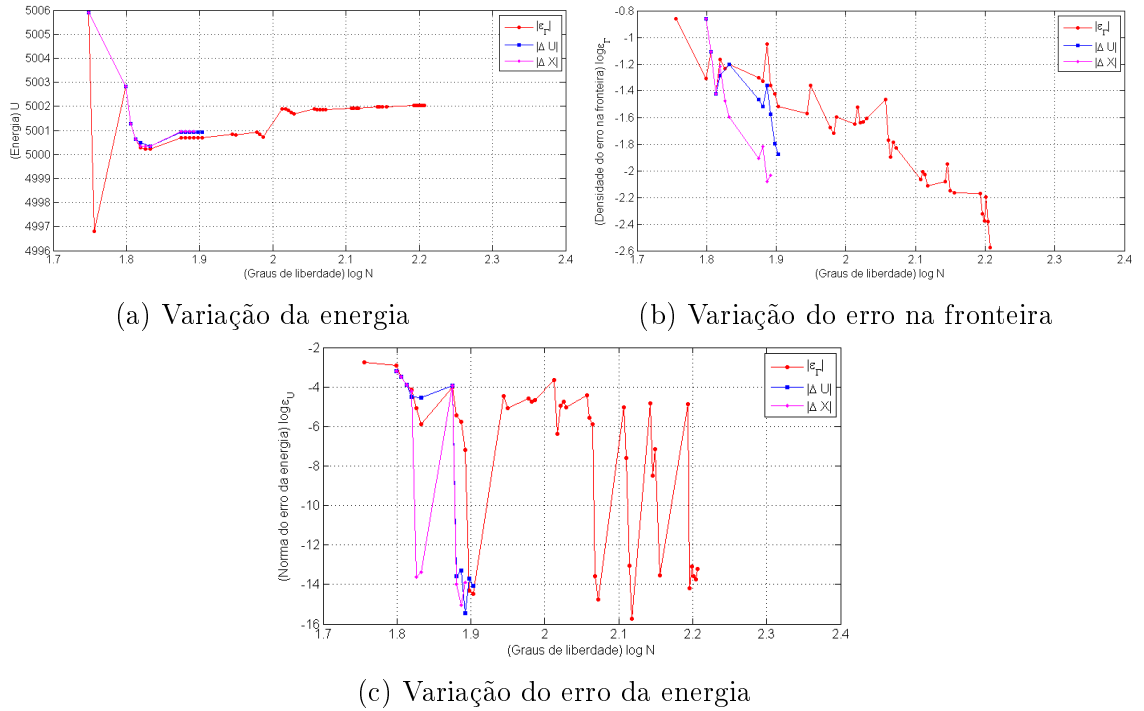


Figura 5.21: Comparação dos três critérios com tolerância de selecção de 5%

Aumentando novamente a tolerância de selecção para 15%, como demonstra a Figura 5.22, as conclusões são semelhantes às obtidas com 5% de tolerância. Novamente, existem discrepâncias entre os valores energéticos de convergência dos três critérios, ainda que reduzidas. Adicionalmente, a medição do tempo computacional revelou que, seleccionar mais fronteiras para cada refinamento não garante uma convergência mais rápida, de facto, neste problema o oposto verifica-se.

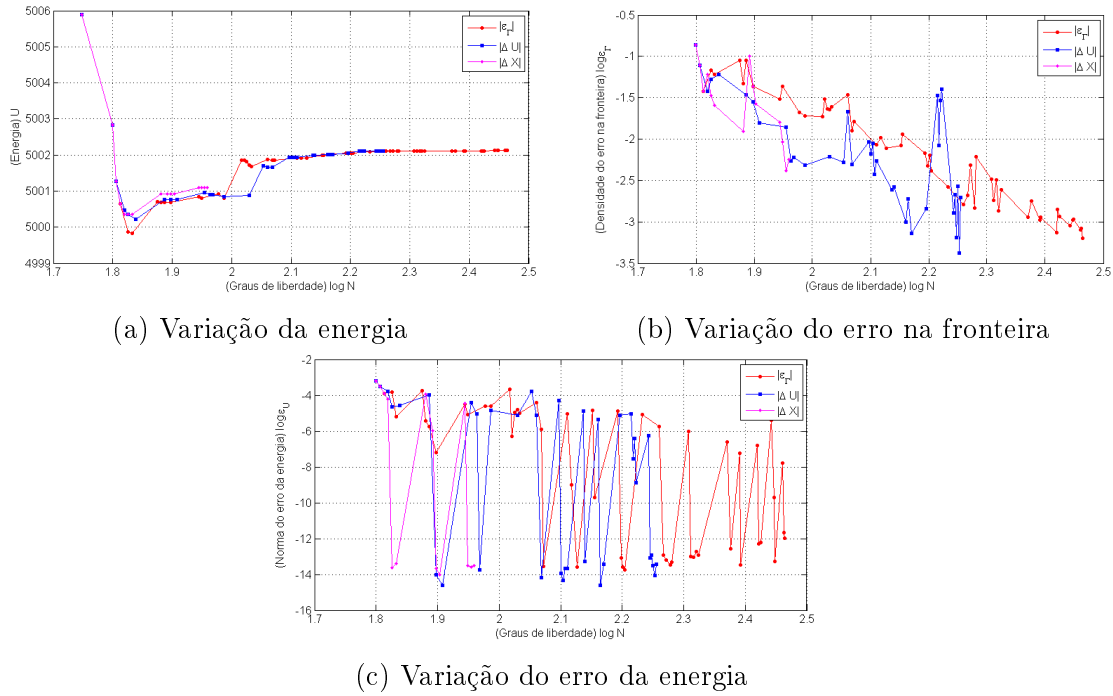


Figura 5.22: Comparação dos três critérios com tolerância de selecção de 15%

A conclusão preliminar é que se demonstra que não existe correspondência directa entre a tolerância de selecção e o número de iterações necessárias para alcançar a convergência, pelo que se recomenda a utilização deste recurso apenas em casos fisicamente fundamentados (e.g. para preservar a simetria de um problema). De um modo geral, o critério ε_T precisa de mais iterações que o critério $\|\Delta X\|$ para conseguir a mesma tolerância de convergência enquanto o critério $\|\Delta U\|$, para resultados semelhantes, exhibe um comportamento intermédio entre os dois mencionados. A possível convergência precoce no critério $\|\Delta X\|$ pode ser controlada aumentando o número de iterações estáveis antes de parar o algoritmo.

Os refinamentos finais em cada fronteira e elementos finitos são apresentados de seguida e confirmam o que os gráficos já sugeriam. O critério $\|\Delta X\|$ tem um comportamento que parece resultar em refinamentos mais localizados. Detecta e refina as fronteiras mais susceptíveis de provocar variações na solução, aumentando rapidamente os graus nestas fronteiras, ao contrário do critério ε_T que tem uma abordagem que causa refinamentos mais uniformemente distribuídos na estrutura.

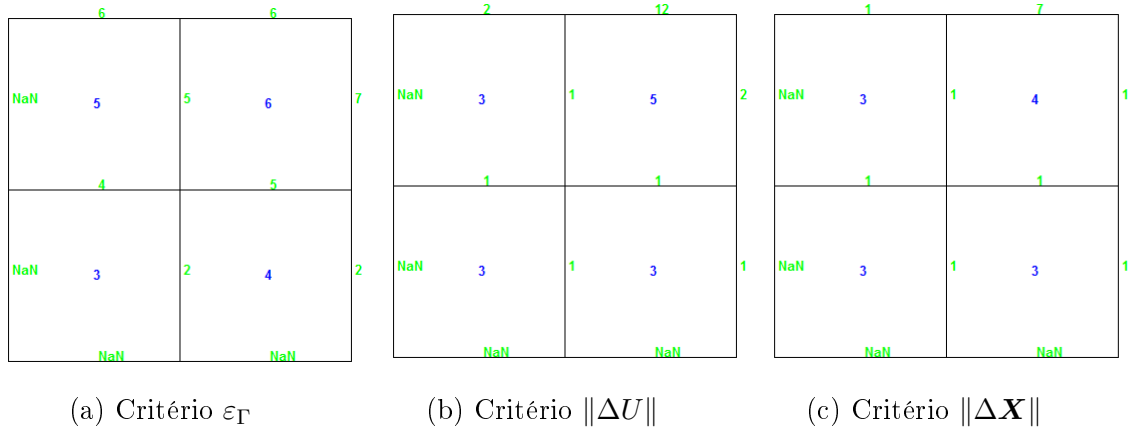


Figura 5.23: Graus finais para diferentes critérios, tolerância de selecção nula

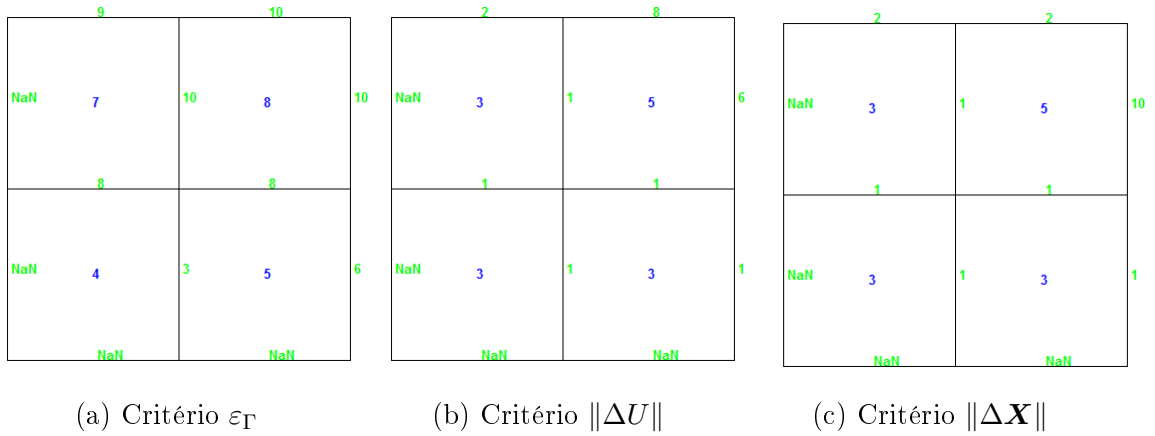


Figura 5.24: Graus finais para diferentes critérios, tolerância de selecção de 5%

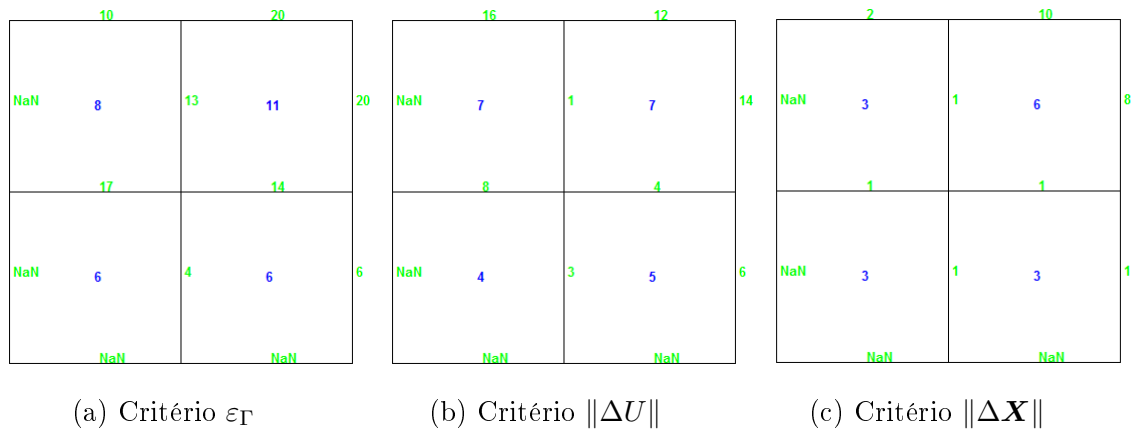


Figura 5.25: Graus finais para diferentes critérios, tolerância de selecção de 15%

5.3 Caso de estudo 3 - problema simétrico

O terceiro caso de estudo analisado é uma variação do caso de estudo 1, desta vez com apenas um eixo de simetria. O problema é uma placa quadrada com condutividade (k) de $1 \text{ W m}^{-1} \text{ K}^{-1}$ e sem geração interna de calor, representado na Figura 5.26a. As condições de fronteira são todas do tipo Dirichlet, com temperaturas impostas (T) de 20°C para a fronteira superior e 0°C para todas as outras fronteiras. O problema foi discretizado numa malha mais refinada de dezasseis elementos, representada na Figura 5.26b.

Este será, à partida, o problema mais exigente do ponto de vista numérico pois é possível identificar-se, nos nós representados a vermelho, singularidades nos campos de fluxos consequência da imposição de duas temperaturas distintas no mesmo ponto. Uma temperatura de 20°C é proveniente da fronteira superior, e outra temperatura de 0°C oriunda da fronteira lateral. As bases de aproximação iniciais admitidas foram, à semelhança dos outros casos de estudo, grau 1 para as fronteiras e grau 3 para os domínios.

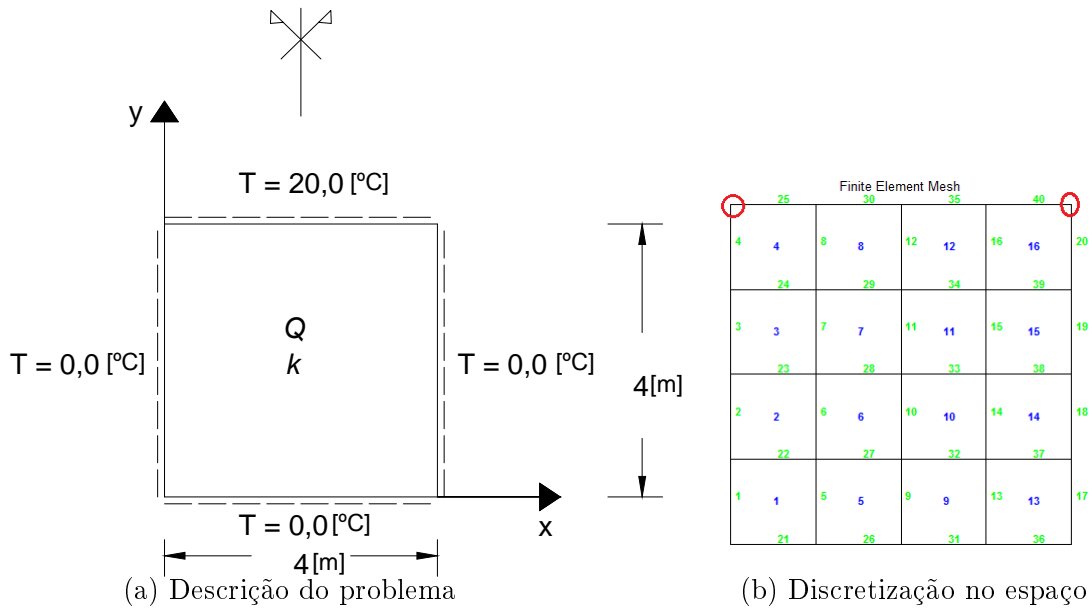


Figura 5.26: Caso de estudo 3 e a malha utilizada na sua discretização

Os resultados obtidos em [41], para o mesmo caso de estudo e a mesma malha são apresentados de seguida. O campo de temperaturas e os graus utilizados na análise são representados nas Figuras 5.27a e 5.27b, respectivamente, e os campos de fluxos estão representados na Figura 5.28.

Observa-se, nos campos de fluxos, uma grande concentração de fluxos nos nós identificados. Estas regiões podem ser problemáticas na resolução do problema, monopolizando a acção do algoritmo com uma preponderância de refinamentos naquelas zonas levando a desequilíbrios no sistema.

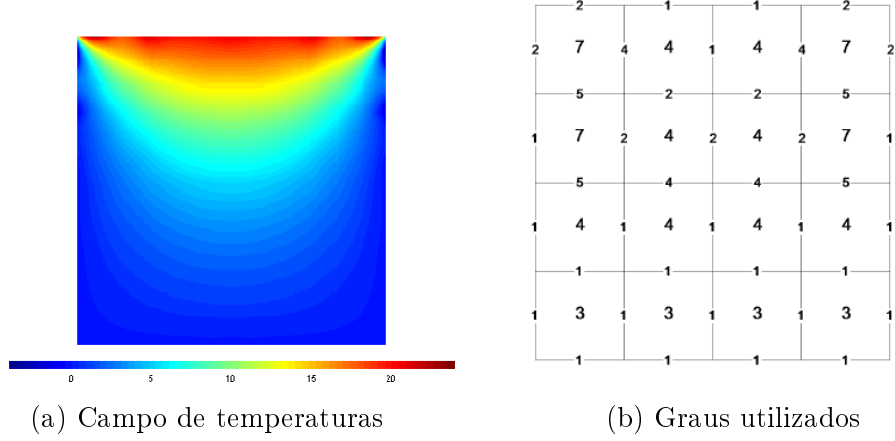


Figura 5.27: Campo de temperaturas e graus das bases de aproximação, em [41]

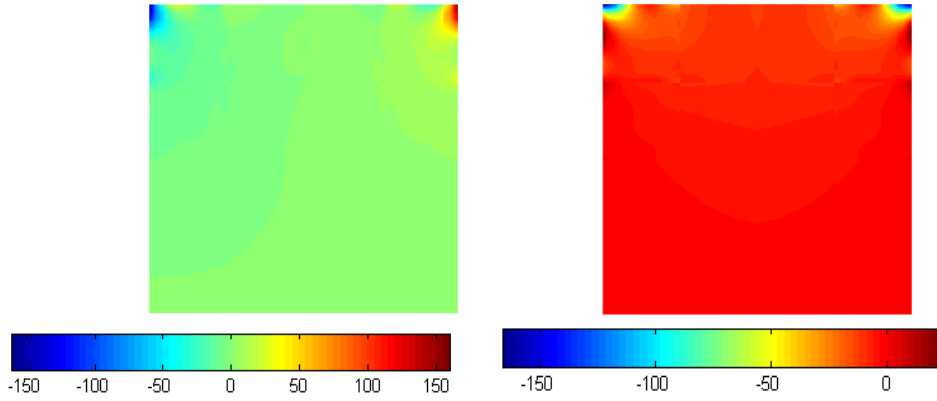


Figura 5.28: Campos de fluxos em [41] (esq q_x , dir q_y)

O caso de estudo será analisado primeiro utilizando como critério de selecção de fronteiras a refinar a densidade do erro na fronteira com 1% e uma tolerância de saída de 10^{-4} . Para este caso de estudo, a tolerância de saída é menos reduzida devido às dificuldades numéricas antevistas. A comparação entre os vários critérios está reservada para posterior estudo após os resultados apresentados.

Novamente, foi representada graficamente a evolução dos parâmetros que condicionam o processo adaptativo: a energia da solução (Figura 5.29), a norma do erro da energia (Figura 5.30), e finalmente, a norma da densidade do erro na(s) fronteira(s) seleccionada(s) para refinamento (Figura 5.31).

A energia demonstra uma evolução irregular durante o processo iterativo, com a ocorrência de alguns aumentos repentinos, alguns de várias ordens de grandeza. Este comportamento demonstra que o algoritmo tem dificuldades neste caso específico com a imposição de diferentes temperaturas num dado ponto. As singularidades têm uma influência clara durante o processo de refinamento, e mesmo na recuperação dos valores de temperatura e fluxos de calor, como se demonstrará mais à frente na representação gráfica dos campos de temperaturas e fluxos. Contudo, a energia converge ao fim de 128 iterações fixando-se em $1.575e + 03$, enquanto o valor obtido

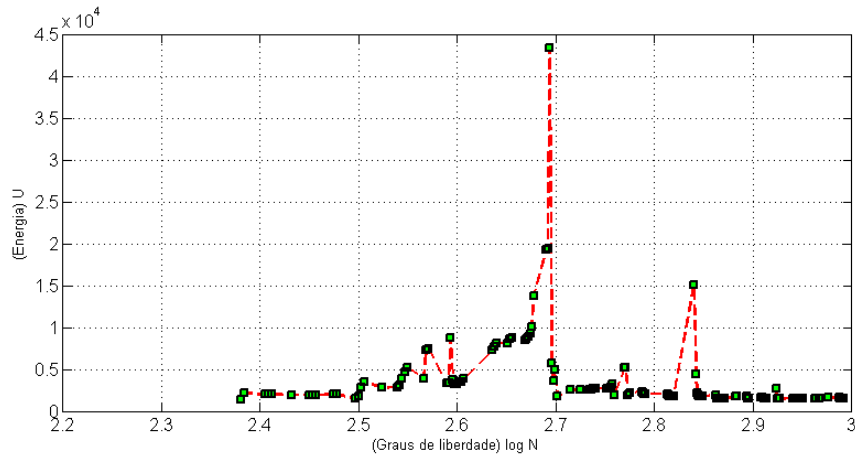


Figura 5.29: Variação da energia com graus de liberdade

em [41] foi de $1.4013e + 03$. A diferença é de aproximadamente 11%, a maior de todos os casos estudados, o que demonstra o comportamento menos competente do algoritmo neste problema.

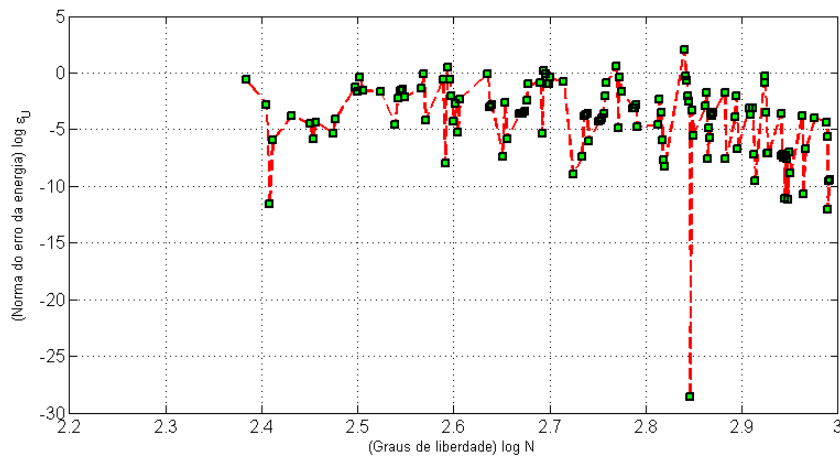


Figura 5.30: Norma do erro da energia no processo iterativo

O gráfico do erro da energia confirma o valor flutuante que a energia toma durante o refinamento. O critério de convergência é cumprido e a tolerância é atingida após várias iterações cujo refinamento não provoca variação significativa do erro da energia. Novamente fica a ideia que o algoritmo não consegue lidar de forma eficiente com o problema, resultando num processo de refinamento deficiente para este tipo de casos de estudo com soluções numericamente mais exigentes.

A densidade do erro na fronteira, em paralelo ao observado com a energia, também exibe uma evolução irregular, ainda que mantendo uma tendência global para diminuir. Uma regressão linear dos erros, a azul, foi adicionada ao gráfico. O coeficiente de determinação R^2 tem o valor de 0.388, um ajuste bastante inferior aos outros casos de estudo, evidenciando a natureza instável do refinamento.

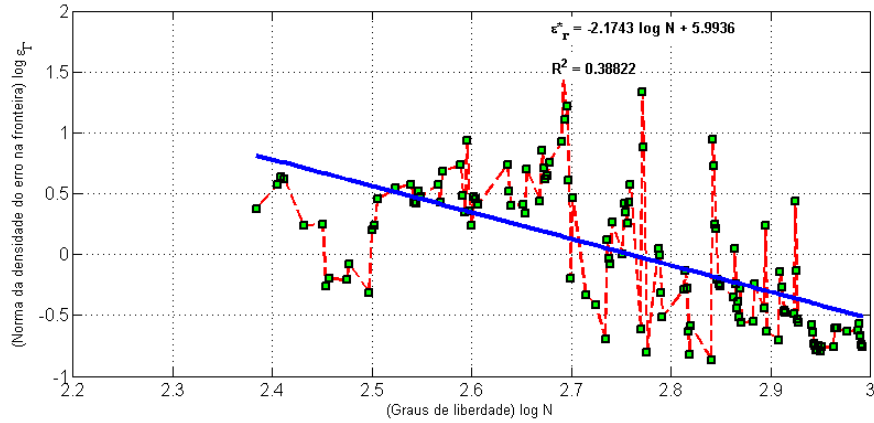


Figura 5.31: Norma da densidade do erro na fronteira no processo iterativo

Os graus das bases de aproximação, no fim do processo de refinamento, estão esquematizados na Figura 5.32, com as bases nas fronteiras a verde claro e as bases nos domínios a azul escuro. O processo iterativo encontra-se listado na Tabela C.3 no Apêndice C.

18		8		8		18	
17	11	19	8	1	8	19	11
18		7		7		18	
8	9	9	6	1	6	9	9
8		6		6		8	
7	7	7	6	1	6	7	7
4		6		6		4	
5	5	5	5	1	5	5	5
2		2		2		2	

Figura 5.32: Graus finais das bases de aproximação

A principal observação é a predominância do refinamento na zona das singularidades o que, como se verá a seguir, até pode ser prejudicial na continuidade inter-elementar do campo de temperaturas. Também é possível observar que existe simetria nos graus finais apresentados. As fronteiras sobre o eixo de simetria não são refinadas à semelhança do caso bisimétrico. De seguida são apresentados os campos de temperatura e de fluxos de calor nas direcções x e y , apenas em alguns passos do processo iterativo, optando-se por apresentá-los de 10 em 10 iterações.

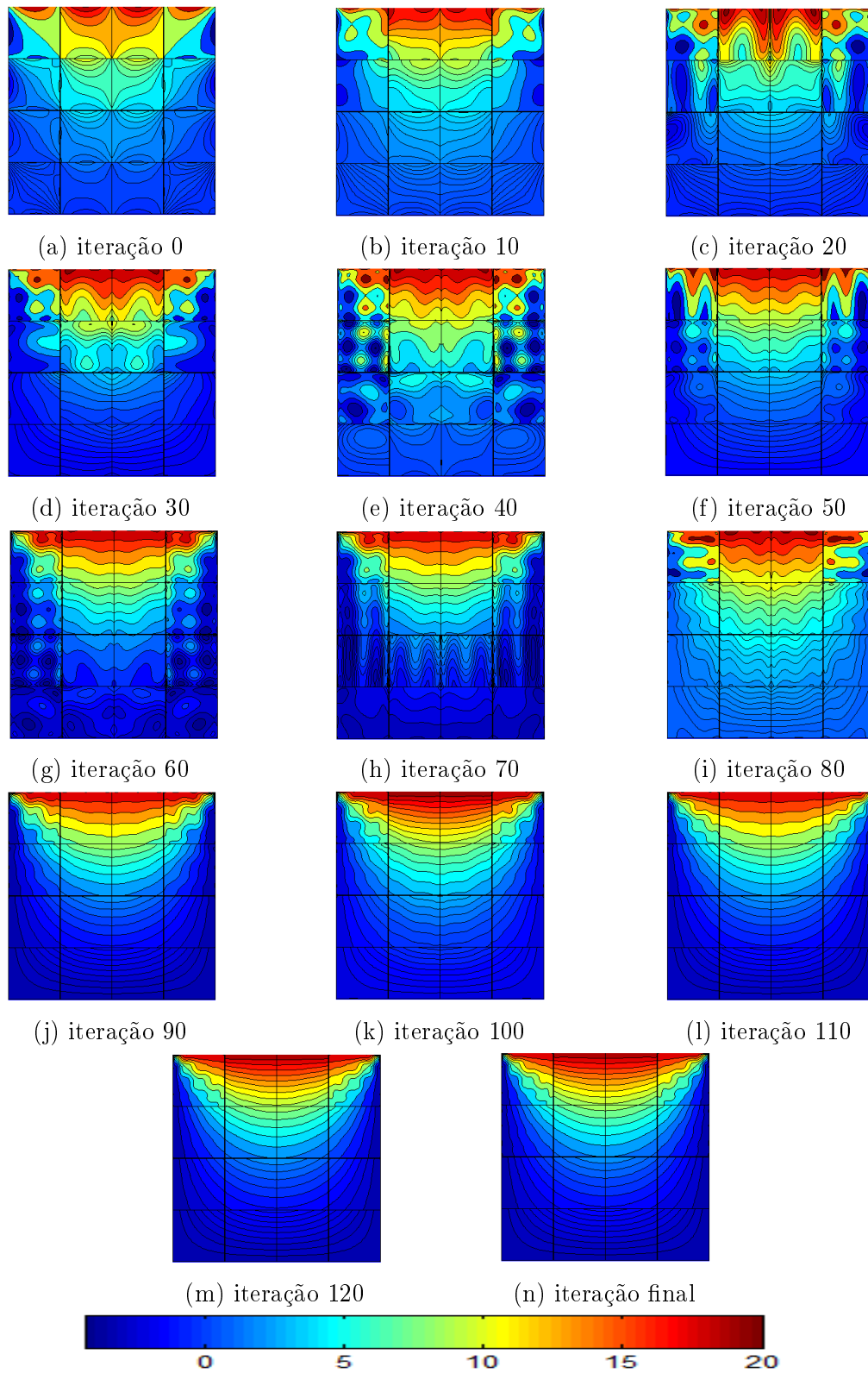


Figura 5.33: Campos de temperaturas

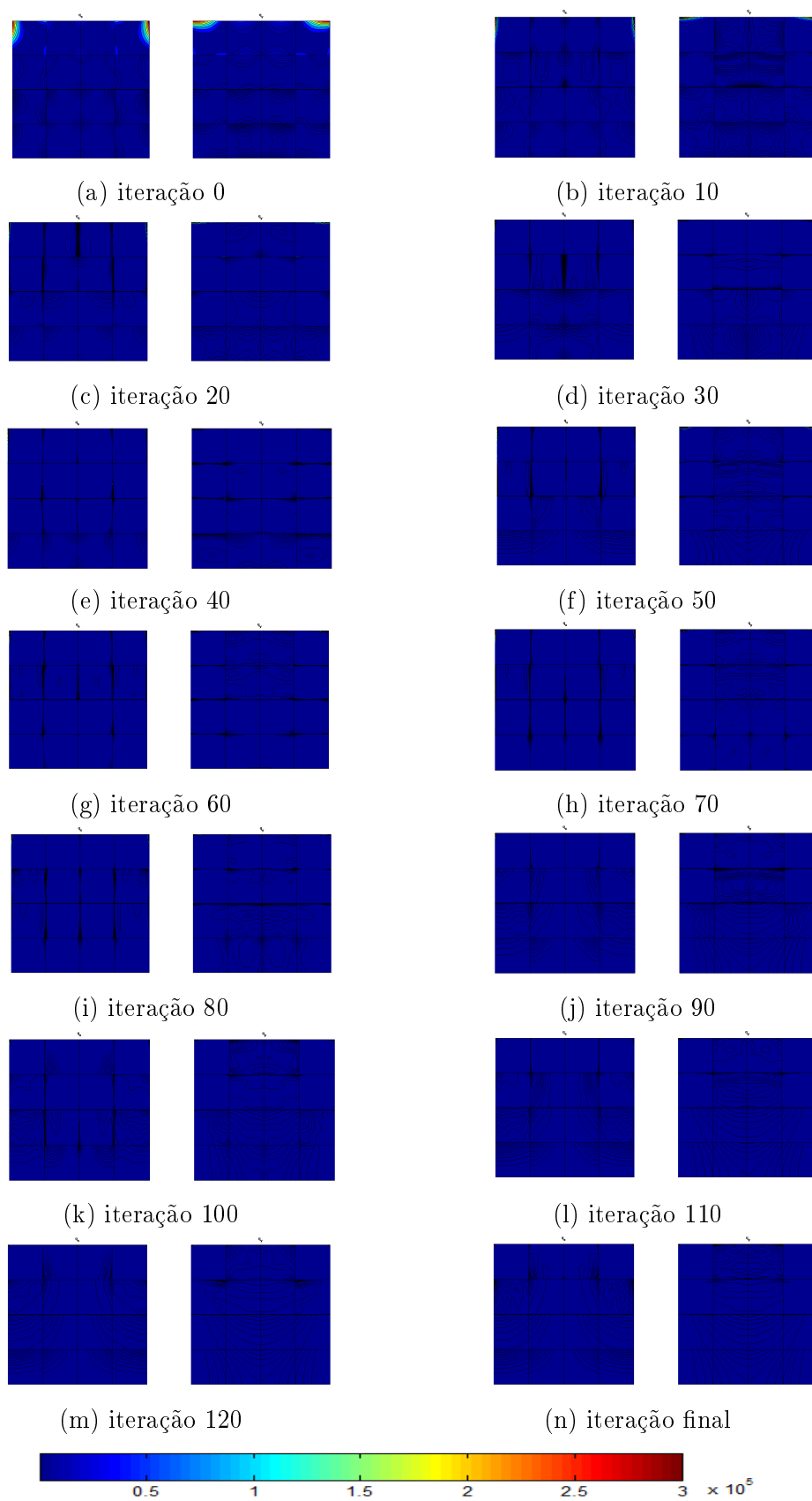


Figura 5.34: Campos de fluxos (esq q_x ; dir q_y)

Como foi referido, este problema tem dois pontos com um alto gradiente de fluxos, consequência de se impôr duas temperaturas no mesmo ponto. Nos campos de fluxos a duas dimensões demonstrados, não é perceptível a sua existência, logo, estes campos para iteração final são representados em três dimensões para ser mais fácil a sua interpretação.

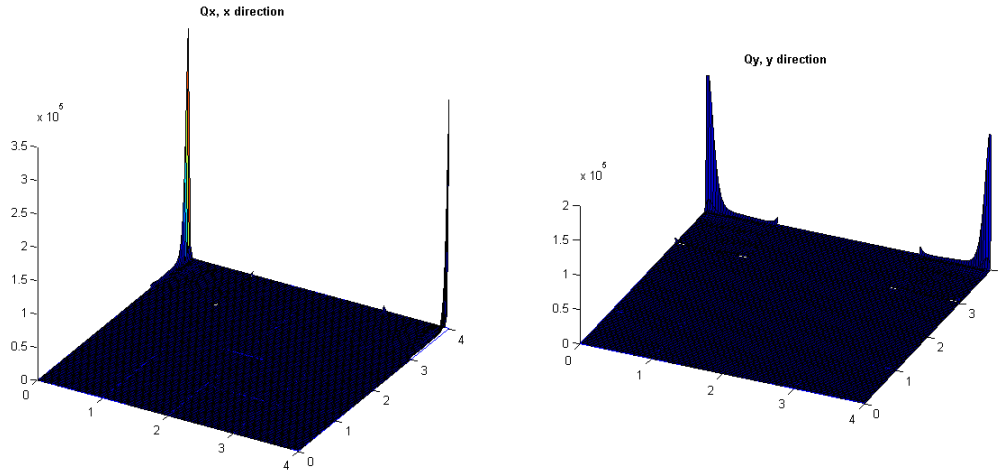


Figura 5.35: Campos de fluxos da iteração final em três dimensões

Observam-se facilmente os dois pontos de singularidade, caracterizados pelo aumento dos fluxos em várias ordens de grandeza de diferença, nos pontos referidos.

Os campos de temperaturas apresentam, tal como a energia, uma evolução inconstante. Algumas iterações exibem campos de temperaturas claramente incoerentes, consequência do comportamento instável do sistema durante o refinamento. No entanto, a partir, aproximadamente, da 100^a iteração o processo já começa a convergir, e os campos de temperaturas são mais equilibrados até atingir a solução final. Os campos de fluxos necessitam de uma representação tridimensional para totalmente se compreender a concentração destes em pontos discretos.

Comparação entre critérios

O caso de estudo 3, como já foi mencionado, apresentou os resultados menos coerentes. Para quaisquer dos critérios de selecção, os resultados foram semelhantes no padrão inconstante que a energia demonstra durante o processo de refinamento com algumas variações abruptas. As variações da energia térmica, da norma do seu erro e da norma da densidade do erro na fronteira estão representados, para os três critérios na Figura 5.36. O critério da densidade do erro na fronteira está representado graficamente a vermelho, o critério da variação de energia a azul e o critério do incremento das temperaturas generalizadas a magenta.

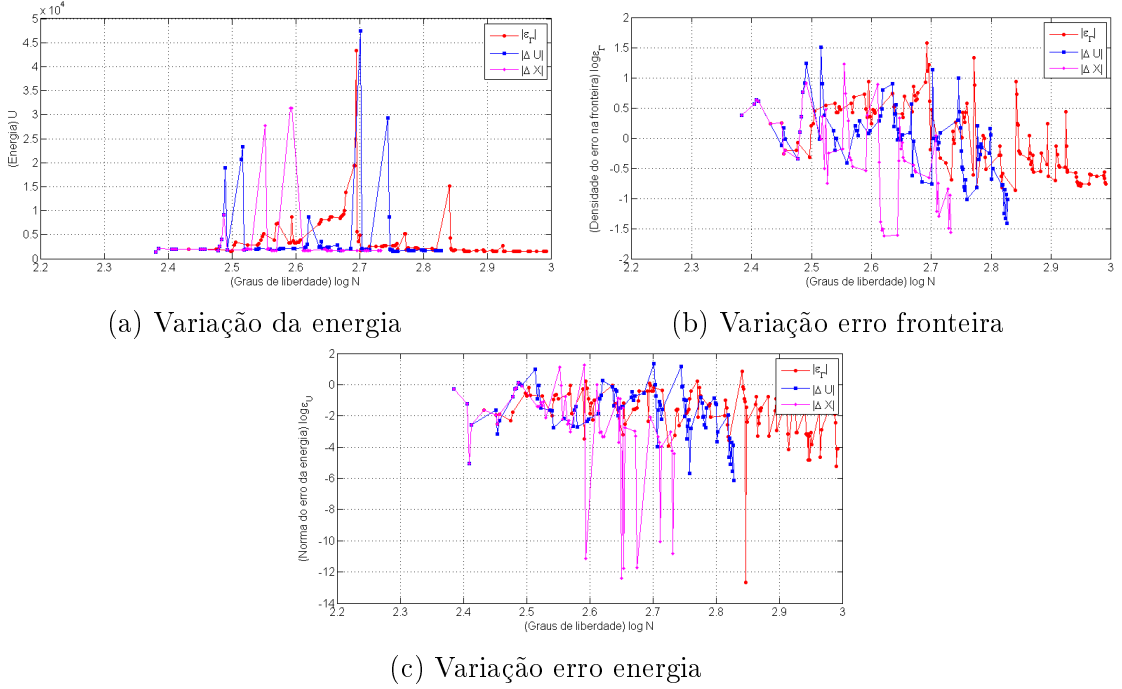


Figura 5.36: Comparação do caso de estudo 3 com tolerância de selecção de 1% e diferentes critérios

Em primeiro lugar, observa-se que todos os critérios chegam à convergência, sendo que as energias térmicas recuperadas no final dos processos iterativos são coerentes. O critério do incremento de temperaturas generalizadas converge em menos iterações que o critério da densidade do erro na fronteira, mas, em certas circunstâncias, corre-se o risco de se obter bases com graus de aproximação desequilibrados, possivelmente prejudicando a qualidade da solução. Os graus obtidos para os diferentes critérios são esquematizados na Figura 5.37 e os campos de temperaturas resultantes no final do processo iterativo na Figura 5.38.

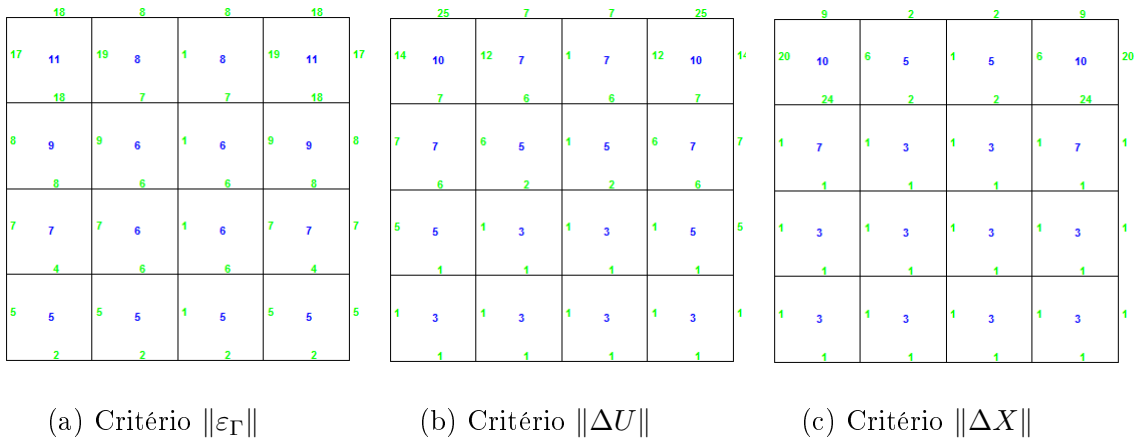


Figura 5.37: Graus finais das bases de aproximação com diferentes critérios

Os resultados obtidos, para qualquer dos critérios adoptados, foram, no mínimo, razoáveis. Ainda assim são perceptíveis ligeiras diferenças entre as soluções, princi-

palmente na continuidade elementar, onde o critério da densidade do erro na fronteira apresentou o campo de temperaturas mais equilibrado, mas também o processo de refinamento mais moroso. Uma análise aos graus finais e respectivos campos de temperaturas comprova a hipótese que uma escolha de graus mais criteriosa fornece resultados mais adequados.

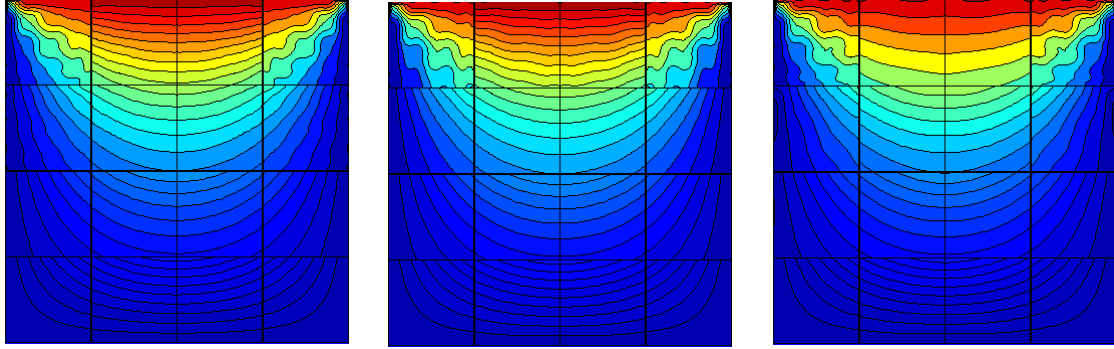
(a) Critério $\|\varepsilon_\Gamma\|$ (b) Critério $\|\Delta U\|$ (c) Critério $\|\Delta \mathbf{X}\|$

Figura 5.38: Campos de temperaturas obtidos com diferentes critérios

Capítulo 6

Conclusões

6.1 Conclusões

O presente trabalho teve como objectivo o desenvolvimento de uma técnica de refinamento para a formulação híbrida do modelo de temperaturas do MEF para problemas de transferência de calor por condução, em duas dimensões, e a sua implementação num algoritmo p -adaptativo automatizado e a validação do mesmo através de um conjunto de testes. A materialização do objectivo foi realizada dentro de uma plataforma implementada em ambiente MATLAB no âmbito desta dissertação. A plataforma visa auxiliar o desenvolvimento de formulações não-convencionais de elementos finitos. Foram criados módulos de pré- e pós-processamento com interface gráfica. Para validar o desempenho da plataforma foi escolhida a formulação híbrida do modelo de temperaturas. O algoritmo p -adaptativo foi inserido na fase de pré-processamento para auxiliar a escolha dos graus ‘óptimos’ das bases de aproximação nos domínios e nas fronteiras da malha dos elementos finitos por um utilizador com pouca experiência na referida formulação. Nesta formulação são então empregues bases de aproximação hierárquicas para os campos das temperaturas nos domínios e dos fluxos nas fronteiras dos elementos, o que torna o refinamento p -adaptativo mais eficiente do que o refinamento h -adaptativo. De modo a avaliar o desempenho do algoritmo foram analisados alguns exemplos apresentados em [41]; um caso bisimétrico, um caso assimétrico e um simétrico, com diferentes tipos de condições de fronteira e características materiais.

A eficiência e robustez do algoritmo foi testada para casos com particularidades onde a implementação computacional poderia ser comprometida. Casos com simetria e/ou com sobreposição pontual de condições de fronteira, exibindo um comportamento satisfatório para dois de três casos, permitindo obter soluções em tempos viáveis. Para o caso de estudo 3 o algoritmo não foi capaz de lidar de forma eficiente, obtendo uma solução razoável mas num processo de refinamento ineficaz e

moroso.

Provou-se a eficácia do algoritmo para problemas simétricos, com um número reduzido de elementos. O algoritmo foi testado para malhas de 16 elementos, com um número elevado de variáveis, num problema com singularidades. Nesse caso, o desempenho do algoritmo ficou aquém do verificado nos primeiros dois casos, o que se deve à existência de fortes gradientes de fluxos nos nós onde as singularidades se encontram. Os resultados permitem aferir que a escolha do critério na selecção de fronteira e o número de quantas fronteiras devem ser refinadas têm uma influência directa no processo de refinamento. Para os casos analisados neste trabalho, conclui-se que uns critérios atingem a convergência de forma mais expedita que outros.

Finalmente, conclui-se que a criação e desenvolvimento de um ambiente gráfico permite uma mais fácil utilização e até um certo ponto, a disseminação da ferramenta, visto que o utilizador não fica sujeito a ter conhecimentos prévios de programação para a introdução de dados. O desenvolvimento desta ferramenta constitui uma excelente base para o seu desenvolvimento, pois permite fáceis adaptações para outros tipos de formulações e algoritmos *p*-adaptativos.

6.2 Desenvolvimentos Futuros

Findo o estudo proposto nesta dissertação, ainda existe uma multitude de direcções a seguir e caminhos a trilhar para o contínuo aperfeiçoamento do método. Um dos primeiros e óbvio a escolher será o efectuar pequenas alterações nesta versão do algoritmo que permitem utilizar outros critérios de selecção de fronteiras ou domínios, e comparar os resultados. O estudo sobre este aspecto foi bastante superficial e apenas se tiraram ilações preliminares, devendo este aspecto ser abordado futuramente de modo a obter uma ideia mais fundamentada. Em versões futuras do algoritmo, implementar a variante ‘completa’, com a listagem completa de refinamentos de fronteiras e elementos, em vez de apenas listar as fronteiras, como na presente versão. Esta alternativa fornecerá processos iterativos de refinamento mais exaustivos e potencialmente, resultados mais precisos. Outros desenvolvimentos futuros serão, por exemplo, integrar a formulação híbrida-Trefftz, e também, adaptar este método com algoritmo adaptativo a *problemas não-lineares*. Problemas de transferência de calor por convecção e radiação inserem-se nesta categoria. Existem inúmeros casos onde estes fenómenos ocorrem em aplicações da vida real, por exemplo, durante um incêndio num prédio. Ainda que a previsão do comportamento de incêndios seja um problema extremamente complexo, a modelação espacial da progressão de um incêndio é uma possível aplicação de problema não-linear. A adaptação para problemas transientes, cuja natureza varie com o tempo, também deve ser abordada de forma a completar mais o espectro de capacidades presentes no programa.

Bibliografia

- [1] J.H. Argyris and S. Kelsey. *Energy Theorems and Structural Analysis*. Butterworth, London, 1960.
- [2] I. Babuška and C. Rheinboldt. A posteriori error estimates for the finite element method. *International Journal of Numerical Methods in Engineering*, 12:1597–615, 1978.
- [3] I. Babuška and C. Rheinboldt. Adaptive approaches and reliability estimates in finite element analysis. *Computer Methods in Applied Mechanics Engineering*, 17/18:519–40, 1979.
- [4] I. Babuška and M. Suri. The p- and hp- versions of the finite element method: An Overview. *Computer Methods in Applied Mechanics and Engineering*, 80:5–26, 1990.
- [5] I. Babuška, B.A. Szabó, and I.N. Katz. The p-version of the finite element method. *SIAM Journal Numer. Anal.*, 18(3):515–545, 1981.
- [6] R.E. Bank. Hierarchical Bases and the Finite Element Method. *Acta Numerica*, 5:1–43, 1996.
- [7] R.E. Bank and T. Dupont. An Optimal Order Process for Solving Finite Element Equations. *Math. Comp.*, 36:35–51, 1981.
- [8] K.J. Bathe. *Finite Element Procedures*. Prentice-Hall, 1996.
- [9] Y.K. Cheung, W.G. Jin, and O.C. Zienkiewicz. Direct solution procedure for solution of harmonic problems using complete, non-singular, Trefftz functions. *Communications in Applied Numerical Methods*, 5:159–169, 1989.
- [10] C. Cismaşiu. *The Hybrid-Trefftz Displacement Element for Static and Dynamic Structural Analysis Problems*. Phd thesis, Instituto Superior Técnico - Universidade Técnica Lisboa, 2000.

- [11] I. Cismaşiu and J.P.M. Almeida. The use of topological data-structures in the distributed solution of non-conventional finite element formulations. *Computers & Structures*, 82:1405–1411, 2004.
- [12] I. Cismaşiu and J.P.M. Almeida. Data structures for the distributed iterative solution of non-conventional finite element models. *Advances in Engineering Software*, 38:750–762, 2007.
- [13] I. Cismaşiu, J.P.M. Almeida, L.M.S. Castro, and D. Harbis. *Parallel solution techniques for hybrid mixed finite element models*. M. Papadrakakis, B.H.V. Topping, (Editors), Innovative Computational Methods for Structural Mechanics, Saxe-Coburg Publications, Stirlingshire, UK, Chapter 6, pp 109-128 edition, 1999.
- [14] R.W. Clough and E.L. Wilson. Early Finite Element Research at Berkeley. *Fifth U.S. National Conference on Computational Mechanics*, 1999.
- [15] R.D. Cook, D.S. Malkus, M.E. Plesha, and R.J. Witt. *Concepts and Applications of Finite Element Analysis*. John Wiley and Sons, fourth edition, 2002.
- [16] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bulletin of the American Mathematical Society*, 49:1–61, 1943.
- [17] A.S. Franca and K. Haghighi. Adaptive finite element analysis of transient thermal problems. *Numerical Heat Transfer, Part B*, 26(3):274–294, 1994.
- [18] J.A.T. Freitas. Hierarchical 2-D Equilibrium Finite Element based on Chebyshev Interpolation Functions. In B.H.V. Topping, editor, *Development in Computational Engineering Mechanics*, pages 115–120. Civil Comp Press, Edingburgh, UK, 1993.
- [19] J.A.T. Freitas, J.P.M. Almeida, and E.M.B.R. Pereira. Non-conventional formulations for the finite element method. *Computational Mechanics*, 23:488–501, 1999.
- [20] J.A.T. Freitas and C. Cismaşiu. Adaptive p -refinement of hybrid-Trefftz finite element solutions. *Finite Elements in Analysis and Design*, 39:1095–1121, 2003.
- [21] H.C. Huang and R.W. Lewis. Adaptive analysis for heat flow problems using error estimation techniques. *Sixth International Conference for Numerical Methods in Thermal Problems*, pages 1029–1044, 1989.
- [22] H.C. Huang and A.S. Usmani. *Finite Element Analysis for Heat Transfer*. Springer London, 1994.

-
- [23] The MathWorks Inc. *MATLAB*[®] Creating Graphical User Interfaces, 2014.
- [24] A.H. Jesus. Análise estrutural com elementos finitos híbridos-Trefftz e refinamento p -adaptativo. Tese mestrado, Faculdade Ciências e Tecnologia - Universidade Nova Lisboa, 2010.
- [25] A.H. Jesus, I. Cismaşiu, and J.A.T. Freitas. Fully automatic p -adaptive hybrid-Trefftz displacement elements. *Computer Methods in Mechanics*, 2011. Warsaw, Poland.
- [26] J. Jirousek and Q.H. Qin. Application of hybrid-Trefftz element approach to the transient heat conduction analysis. *Computers & Structures*, 58:195–201, 1996.
- [27] D.W. Kelly, J.P. De S.R. Gago, and O.C. Zienkiewicz. A-posteriori error analysis and adaptive processes in the finite element method: Part I - error analysis. *International Journal for Numerical Methods in Engineering*, 19:1593–1619, 1983.
- [28] D.W. Kelly, J.P. De S.R. Gago, and O.C. Zienkiewicz. A-posteriori error analysis and adaptive processes in the finite element method: Part II - adaptive mesh refinement. *International Journal for Numerical Methods in Engineering*, 19:1621–1656, 1983.
- [29] L-E. Lindgren. From Weighted Residual Methods to Finite Element Methods, 2009.
- [30] R. Mahnken and E. Stein. Adaptive time-step control in creep analysis. *International Journal for Numerical Methods in Engineering*, 28:1619–1633, 1989.
- [31] MathWorks. *MATLAB*[®] from mathworks. www.mathworks.com.
- [32] I.D. Moldovan. *Hybrid-Trefftz Elements for Elastodynamic Analysis of Saturated Porous Media*. Phd thesis, Instituto Superior Técnico - Universidade Técnica Lisboa, 2007.
- [33] G. Nikishov. *Programming Finite Elements in Java*TM. Springer-Verlog London, first edition, 2010.
- [34] J.T. Oden, W. Wu, and M. Ainsworth. An a-posteriori error estimate for FE approximations of the Navier-Stokes equations. *TICOM Report, TR93-05*, 1993.

- [35] D.H. Pelletier and J. Hetu. An adaptive finite element methodology for incompressible viscous flow. *Advances in FEA in Fluid Dynamics*, pages 1–11, 1992.
- [36] O.J.B.A. Pereira. *Utilização de Elementos Finitos de Equilíbrio em Refinamento Adaptativo*. Tese doutoramento, Instituto Superior Técnico - Universidade Técnica Lisboa, 1996.
- [37] O.J.B.A. Pereira, J.P.M. Almeida, and E.A.W. Maunder. Adaptive methods for hybrid equilibrium finite element models. *Computer Methods in Applied Mechanics and Engineering*, 176:19–39, 1999.
- [38] M.A. Piteri. *Geração automática de malhas hierárquico-adaptativas em domínios bidimensionais e tridimensionais*. Tese doutoramento, Instituto Superior Técnico - Universidade Técnica Lisboa, 1998.
- [39] J.N. Reddy. *An Introduction to the Finite Element Method*. McGraw-Hill, second edition, 1993.
- [40] M.K. Sangare, E. Danho, R.N. Djue, and K.E. Kanga. Comparative study of different hierarchical bases of finite element method: Application to elastostatic analysis of two-dimensional structures. *Asian Journal of Applied Sciences*, 2:139–149, 2009.
- [41] C.R.S. Silva. Desenvolvimento de formulações de elementos finitos para problemas de transferência de calor. Tese mestrado, Faculdade Ciências Tecnologia - Universidade Nova Lisboa, 2014.
- [42] J.L. Synge. Triangulation in the Hypercircle Method for Plane Problems. *Proc. R. Irish Acad.*, Vol. 54A21, 1952.
- [43] J. Wu, J.Z. Zhu, and O.C. Zienkiewicz. Error estimation and adaptivity in flow formulation for forming problems. *Computer Mechanics*, 6:259–270, 1990.
- [44] J.Z. Zhu and O.C. Zienkiewicz. Adaptive techniques in the Finite Element Method. *Communications in Applied Numerical Methods*, 4:197–204, 1988.
- [45] O.C. Zienkiewicz. Origins, milestones and directions of the finite element method - a personal view. *Archives of Computational Methods in Engineering*, 2:1–48, 1995.
- [46] O.C. Zienkiewicz, J.P. De S.R. Gago, and D.W. Kelly. The hierarchical concept in finite analysis. *Computers & Structures*, 16:53–65, 1983.

-
- [47] O.C. Zienkiewicz, Y.C. Liu, and G.C. Huang. Error estimation and adaptivity in flow formulation for forming problems. *International Journal for Numerical Methods in Engineering*, 25:2191–2202, 1988.
 - [48] O.C. Zienkiewicz and D.V. Phillips. An automatic mesh generation scheme for plane and curved surfaces by ‘isoparametric’ co-ordinates. *International Journal for Numerical Methods in Engineering*, 3:519–528, 1971.
 - [49] O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The Finite element Method. Its Basis and Fundamentals*. Elsevier Butterworth Heinemann, sixth edition, 2005.
 - [50] O.C. Zienkiewicz and J.Z. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering*, 24:337–357, 1987.

Apêndice A

Equação governativa do problema na forma fraca

A equação governativa diferencial no domínio representa a forma forte do problema, dada pela seguinte expressão,

$$\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + Q = 0 \quad (\text{A.1})$$

com as condições de fronteira de *Dirichlet*

$$T(x, y) = T_{\Gamma}(x, y) \quad \text{em } \Gamma_u \quad (\text{A.2})$$

e condições de fronteira de *Neumann*

$$n q(x, y) = q_{\Gamma}(x, y) \quad \text{em } \Gamma_{\sigma}. \quad (\text{A.3})$$

Visto que a forma forte (A.1) não tem, em geral, soluções analíticas que satisfaçam as condições de fronteira (A.2) e (A.3), o problema é resolvido utilizando métodos numéricos aproximados cuja formulação é baseada no método dos resíduos pesados.

Sendo \bar{T} a aproximação de T do problema, o resíduo do método define-se como:

$$R = \frac{\partial}{\partial x} \left(k \frac{\partial \bar{T}}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial \bar{T}}{\partial y} \right) + Q \quad \text{em } \Omega \quad (\text{A.4})$$

A forma fraca do problema toma então a forma,

$$\int_{\Omega} w R d\Omega = 0 \quad (\text{A.5})$$

onde $w(x, y)$ é a função (arbitrária) de peso. Substituindo a definição de resíduo na equação (A.5), obtém-se:

$$\int_{\Omega} w \left[\frac{\partial}{\partial x} \left(k \frac{\partial \bar{T}}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial \bar{T}}{\partial y} \right) + Q \right] d\Omega = 0 \quad (\text{A.6})$$

Integrando por partes a equação (A.6), separando a parte do domínio da parte da fronteira, de acordo com o Teorema de Green, obtém-se

$$\int_{\Omega} w \left[\frac{\partial}{\partial x} \left(k \frac{\partial \bar{T}}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial \bar{T}}{\partial y} \right) \right] d\Omega = - \int_{\Omega} \left[\frac{\partial w}{\partial x} \left(k \frac{\partial \bar{T}}{\partial x} \right) + \frac{\partial w}{\partial y} \left(k \frac{\partial \bar{T}}{\partial y} \right) \right] d\Omega \quad (\text{A.7})$$

$$+ \int_{\Gamma} w k \frac{\partial \bar{T}}{\partial n} d\Gamma$$

Separando o último termo da expressão (A.7) nas partes de Neumann e Dirichlet, é possível impor imediatamente a condição (A.3):

$$- \int_{\Omega} \left(\frac{\partial w}{\partial x} k \frac{\partial \bar{T}}{\partial x} + \frac{\partial w}{\partial y} k \frac{\partial \bar{T}}{\partial y} - w Q \right) d\Omega + \int_{\Gamma} w k \left(\frac{\partial \bar{T}}{\partial x} n_x + \frac{\partial \bar{T}}{\partial y} n_y \right) d\Gamma_u \quad (\text{A.8})$$

$$- \int_{\Gamma_{\sigma}} w (n q_{\Gamma}) d\Gamma_{\sigma} = 0$$

onde se teve em conta que,

$$\Gamma = \Gamma_u \cup \Gamma_{\sigma} \quad (\text{A.9})$$

e

$$k \frac{\partial \bar{T}}{\partial n} = k \left(\frac{\partial \bar{T}}{\partial x} n_x + \frac{\partial \bar{T}}{\partial y} n_y \right) = -n \bar{q} \quad (\text{A.10})$$

A equação (A.8) pode ser simplificada e reescrita da seguinte maneira:

$$\int_{\Omega} (\nabla w)^T k \nabla \bar{T} d\Omega + \int_{\Gamma_u} w n \bar{q} d\Gamma_u + \int_{\Gamma_{\sigma}} w q_{\Gamma} d\Gamma_{\sigma} - \int_{\Omega} w Q d\Omega = 0 \quad (\text{A.11})$$

sendo esta a equação (2.42) presente no texto.

Apêndice B

Exemplo de utilização do software

Neste apêndice é explicado de forma simples, como utilizar o software desenvolvido nesta dissertação. Um exemplo prático é demonstrado e todos os passos a seguir são descritos.

O menu inicial de inserção de dados é presente na Figura B.1.

The screenshot shows a window titled 'exemploGUI' with a standard Windows title bar (minimize, maximize, close buttons). The main content area is divided into two columns. The left column is titled 'Input Values' and contains a paragraph of instructions: 'In this menu several input values should be placed. Some are defined by the material, such as thermal conductivity and the possibility of internal heating. Others are geometric, like the plate's dimensions and number of elements it should be divided, and the rest are for the program's functionality purposes, like the Gauss integration and initial borders and edges order as well.' Below this, there are input fields for 'Thermal Conductivity [W/(m.K)]', 'Internal Heating [W/(m²)] (0 if non-existent)', 'Gauss Integration' (with sub-fields for 'Number of Gauss Intervals' and 'Number of Gauss Points'), and 'Initial Boundary and Domain Order' (with sub-fields for 'Edges Order' and 'Loops Order'). The right column is titled 'Geometric Conditions' and contains input fields for 'Plate Dimensions in X', 'Plate Dimensions in y', 'Element Number in x', and 'Element Number in y'. At the bottom right of the window, there are two buttons labeled 'Reset' and 'Next'.

Figura B.1: Menu inicial

Neste menu, existem diferentes secções para diferentes tipos de dados. Os dados geométricos como as dimensões e número de elementos, em ambas direcções da malha, são colocados na zona delimitada a vermelho da Figura B.2.

The screenshot shows a software window titled 'exemploGUI'. It contains two main sections: 'Input Values' on the left and 'Geometric Conditions' on the right. The 'Input Values' section includes a text block explaining the input requirements, followed by fields for 'Thermal Conductivity [W/(m.K)]' and 'Internal Heating [W/(m²)] (0 if non-existent)'. Below these are 'Gauss Integration' fields for 'Number of Gauss Intervals' and 'Number of Gauss Points'. The 'Geometric Conditions' section, highlighted with a red box, contains fields for 'Plate Dimensions in X', 'Plate Dimensions in y', 'Element Number in x', and 'Element Number in y'. At the bottom of the window are 'Reset' and 'Next' buttons.

Figura B.2: Menu inicial - Dados geométricos

Os graus das bases de aproximação para o domínio e fronteira são introduzidos na zona delimitada na Figura B.3.

This screenshot shows the same 'exemploGUI' window. In this view, the 'Initial Boundary and Domain Order' section is highlighted with a red box. This section contains fields for 'Edges Order' and 'Loops Order'. The 'Geometric Conditions' section is no longer highlighted. The 'Input Values' section and the 'Reset'/'Next' buttons remain visible at the bottom.

Figura B.3: Menu inicial - Graus iniciais das bases de aproximação

A definição da condutividade térmica e da geração interna de calor estão representadas na zona delimitada na Figura B.4.

The screenshot shows a software window titled 'exemploGUI'. It has a tabbed interface with the 'Input Values' tab selected. The window is divided into two main columns. The left column contains a text block explaining input values, followed by two input fields: 'Thermal Conductivity [W/(m.K)]' and 'Internal Heating [W/(m²)] (0 if non-existent)'. These two fields are enclosed in a red rectangular box. Below them are two more input fields: 'Number of Gauss Intervals' and 'Number of Gauss Points'. The right column is titled 'Geometric Conditions' and contains four input fields: 'Plate Dimensions in X', 'Plate Dimensions in y', 'Element Number in x', and 'Element Number in y'. At the bottom of the right column, there are two more input fields: 'Edges Order' and 'Loops Order'. At the very bottom of the window, there are two buttons: 'Reset' and 'Next'.

Figura B.4: Menu inicial - Condutividade térmica e aquecimento interno

Os dados sobre a integração numérica de Gauss utilizada no programa são inseridos na secção delimitada na Figura B.5. Nesta versão do algoritmo é recomendado que se utilize 30 pontos de Gauss e 1 intervalo de Gauss.

This screenshot is similar to the previous one, showing the 'exemploGUI' window with the 'Input Values' tab. In this view, the 'Thermal Conductivity' and 'Internal Heating' fields are no longer highlighted. Instead, the 'Number of Gauss Intervals' and 'Number of Gauss Points' fields in the 'Gauss Integration' section are highlighted with a red rectangular box. The rest of the interface, including the 'Geometric Conditions' section and the 'Reset' and 'Next' buttons, remains the same.

Figura B.5: Menu inicial - Integração numérica

Finalmente, neste menu inicial, referência para os botões no canto inferior direito, demonstrado na Figura B.6, onde o 'Reset' apaga todos os dados inseridos até agora caso o utilizador assim o deseje, e o botão 'Next' que faz prosseguir o GUI, dando acesso ao menu seguinte.

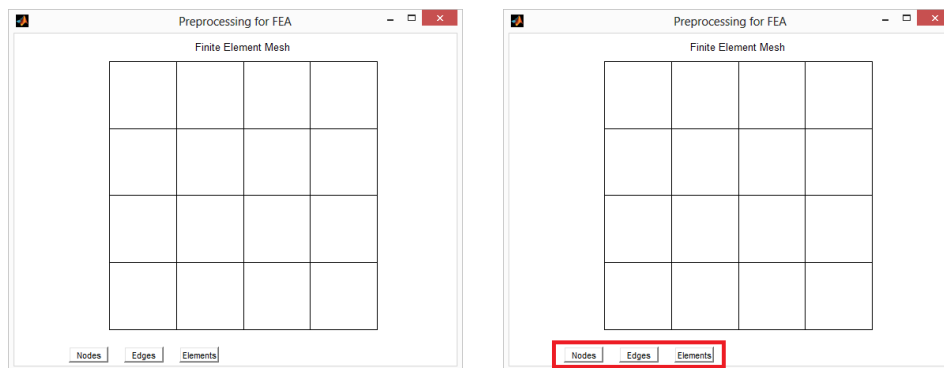
The screenshot shows a window titled 'exemploGUI'. It contains several sections with input fields:

- Input Values:**
 - Thermal Conductivity [W/(m.K)]:
 - Internal Heating [W/(m²)]: (0 if non-existent)
- Geometric Conditions:**
 - Plate Dimensions in X:
 - Plate Dimensions in y:
 - Element Number in x:
 - Element Number in y:
- Gauss Integration:**
 - Number of Gauss Intervals:
 - Number of Gauss Points:
- Initial Boundary and Domain Order:**
 - Edges Order:
 - Loops Order:

At the bottom right, there are two buttons: 'Reset' and 'Next', which are enclosed in a red rectangular box.

Figura B.6: Menu inicial - Botões Reset e Next

No passo seguinte irão aparecer duas janelas, uma com a malha gerada a partir dos dados fornecidos e um menu para definição das fronteiras exteriores de Dirichlet ou Neumann. A malha gerada para um exemplo de 16 elementos está representada na Figura B.7a. A numeração de nós, fronteiras e elementos é activada ou desactivada pelo conjunto de botões presentes na janela, demonstrado na Figura B.7b.



(a) Malha gerada pelos dados inseridos

(b) Botões para activar numeração de nós, fronteiras e elementos

Figura B.7: Malha gerada

O menu das fronteiras está representado na Figura B.8.

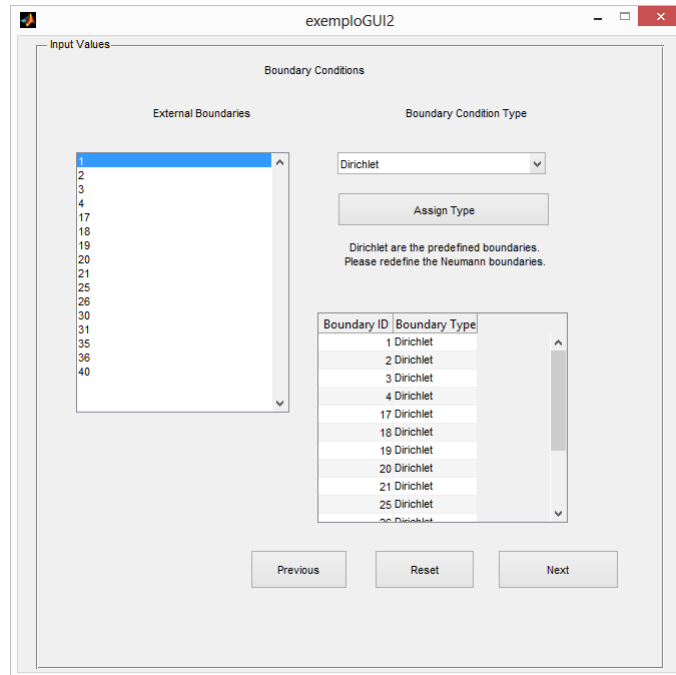


Figura B.8: Menu 2 - Definição das fronteiras exteriores

A listagem das fronteiras exteriores da malha gerada é fornecida na zona delimitada na Figura B.9,

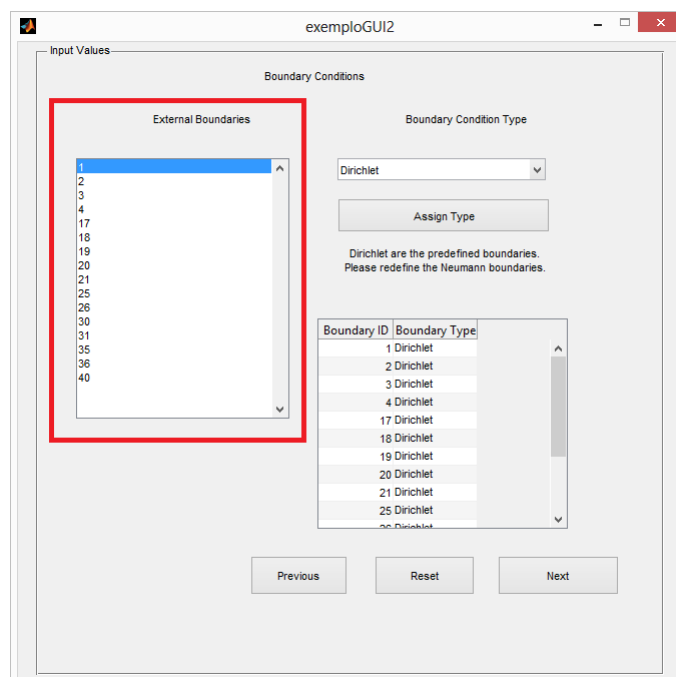


Figura B.9: Menu 2 - Lista de fronteiras exteriores

O utilizador selecciona a fronteira desejada na lista à esquerda e atribui-lhe o tipo desejado. Depois de confirmar a atribuição, o tipo de fronteira é alterado na lista delimitada na Figura B.10.

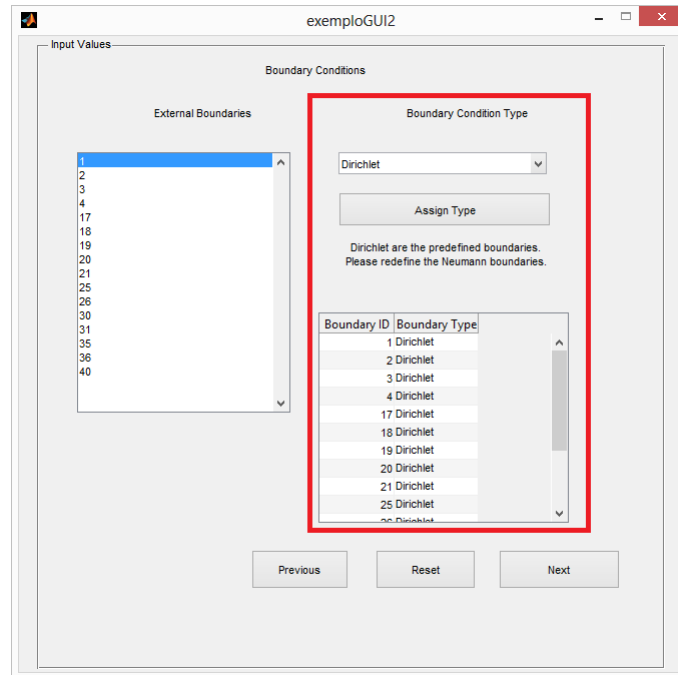


Figura B.10: Menu 2 - Definir tipo de fronteira

O botão 'Previous' retrocede o processo para o menu anterior caso o utilizador queira alterar alguma coisa. O botão 'Reset' reestabelece os tipos iniciais das fronteiras, que por defeito é Dirichlet. O botão 'Next' dá acesso ao menu seguinte.

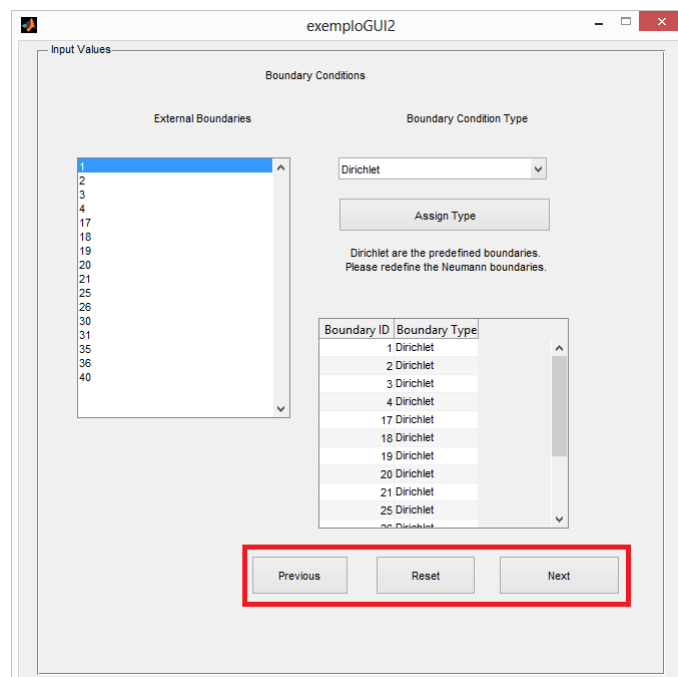


Figura B.11: Menu 2 - Botões Previous, Reset e Next

O terceiro e último menu de inserção de dados é o de definição de condições de fronteira. As fronteiras Dirichlet e Neumann são separadas para o utilizador determinar as temperaturas e os fluxos, respectivamente. Esta janela está representada na Figura B.12.

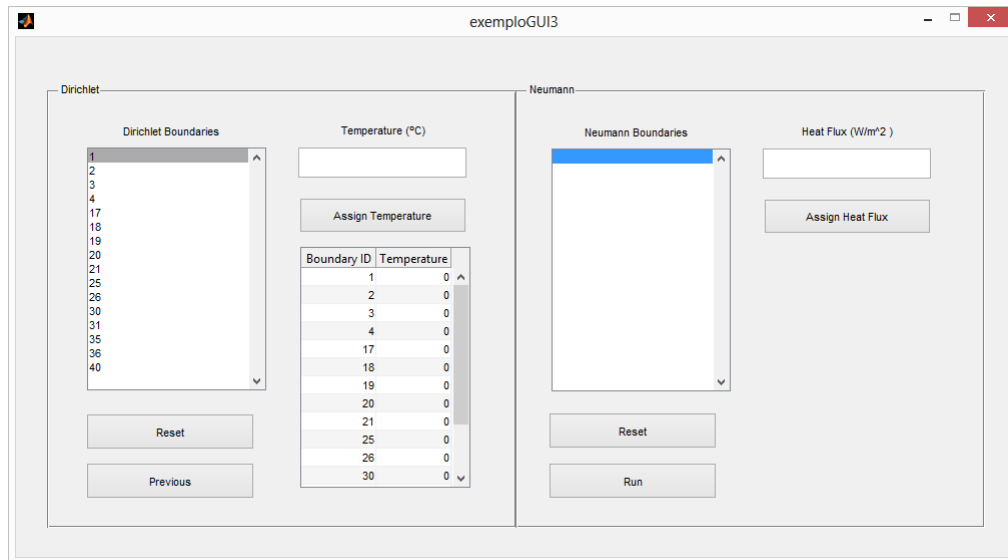


Figura B.12: Menu 3 - Definição das condições de fronteira

As fronteiras são listadas de acordo com o tipo, como está representado na Figura B.13,

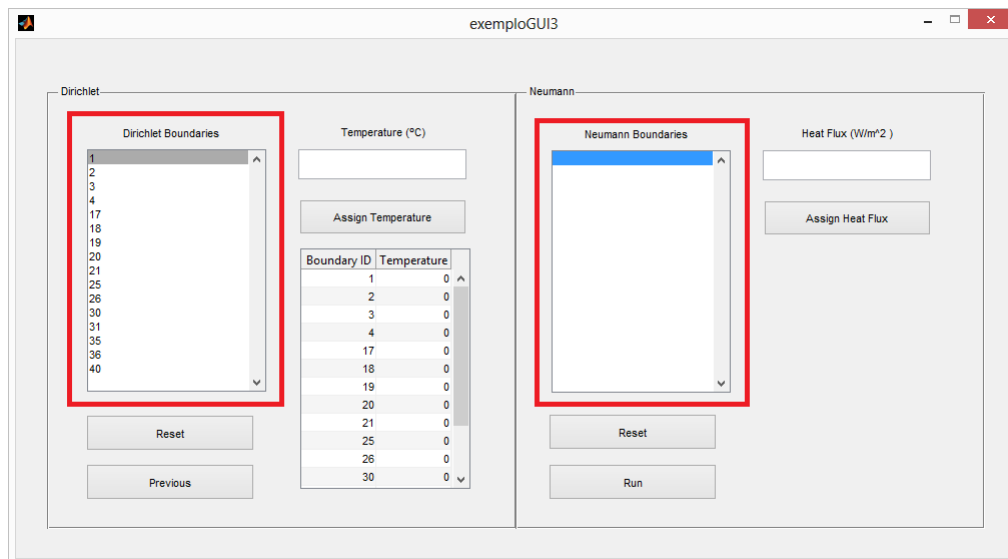


Figura B.13: Menu 3 - Listas de fronteiras Dirichlet e Neumann

Na Figura B.14 está demonstrado onde definir as temperaturas nas fronteiras de Dirichlet e os fluxos nas fronteiras de Neumann, respectivamente.

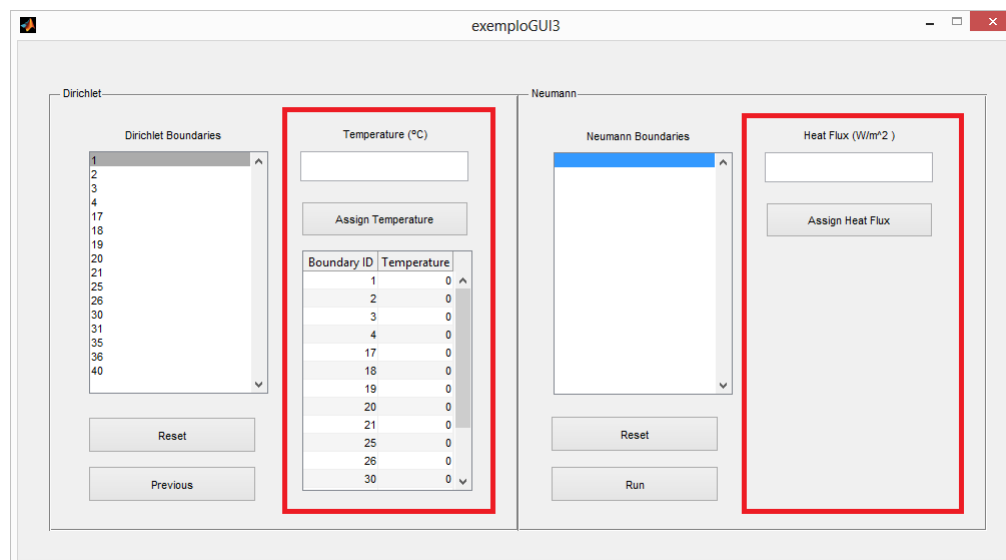


Figura B.14: Menu 3 - Definição das temperaturas e fluxos de calor

Por fim, os botões 'Reset' em cada lista restabelece as condições iniciais, caso o utilizador pretenda alterar alguma já definida. O botão 'Previous', como no menu anterior, dá acesso ao passo anterior caso assim se deseje. O botão 'Run' deve ser clicado apenas quando já todas as condições de fronteira sejam definidas e inicia a análise e consequente refinamento do problema estabelecido. A Figura B.15 representa a posição dos botões acima mencionados.

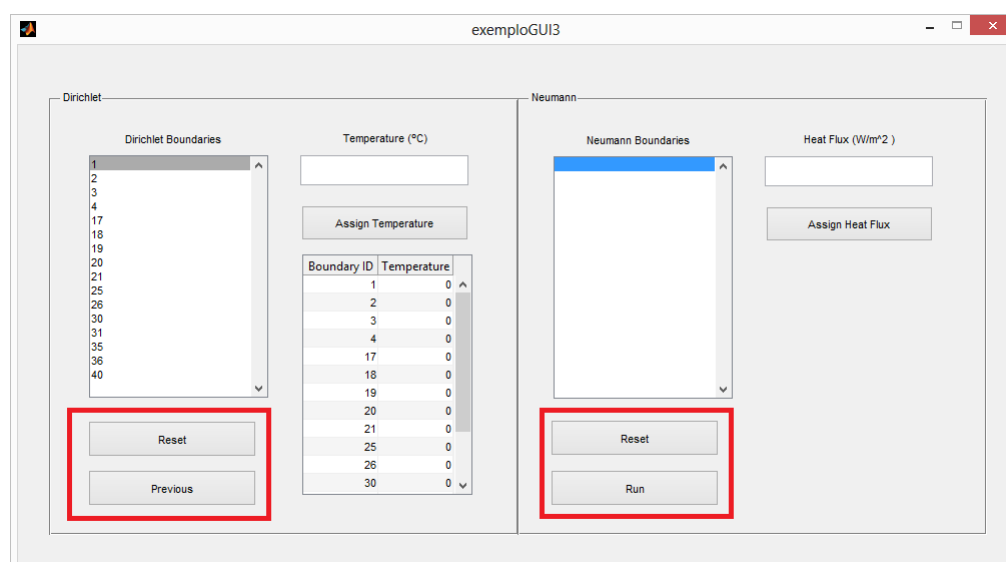


Figura B.15: Menu 3 - Botões Reset, Previous e Run

No início da análise, uma alteração à janela da malha gerada é efectuada, um botão que mostra as condições de fronteira é adicionado, de maneira a garantir de uma forma visual, que está de acordo com os dados inseridos. Na Figura B.16 está representado o novo botão

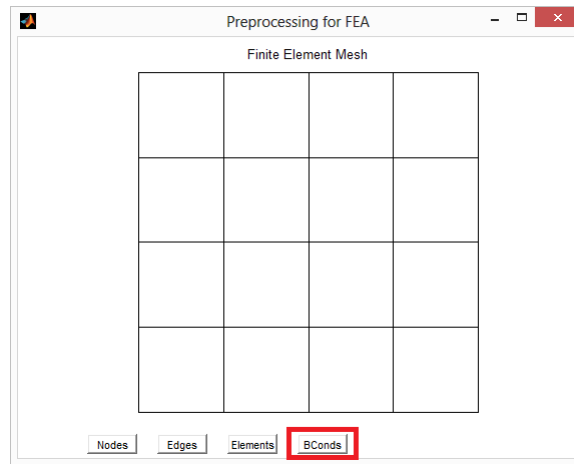


Figura B.16: Malha - Botão de condições de fronteira

Após o processo automático de refinamento adaptativo, são retornadas diversas janelas com os resultados do programa. A primeira janela apresenta o campo de temperaturas na peça. Esta janela está representada na Figura B.17, exposta a seguir.

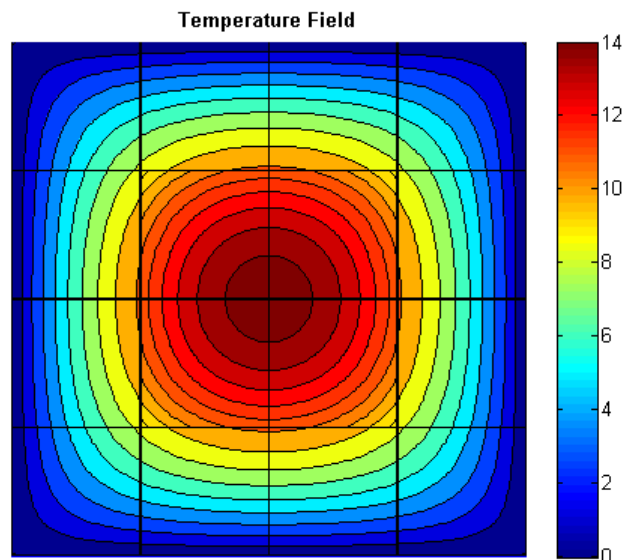


Figura B.17: Exemplo de campo de temperaturas

A janela seguinte (Figura B.18) tem duas imagens, uma para o campo de fluxos na direcção horizontal e outra para o campo de fluxos na direcção vertical.

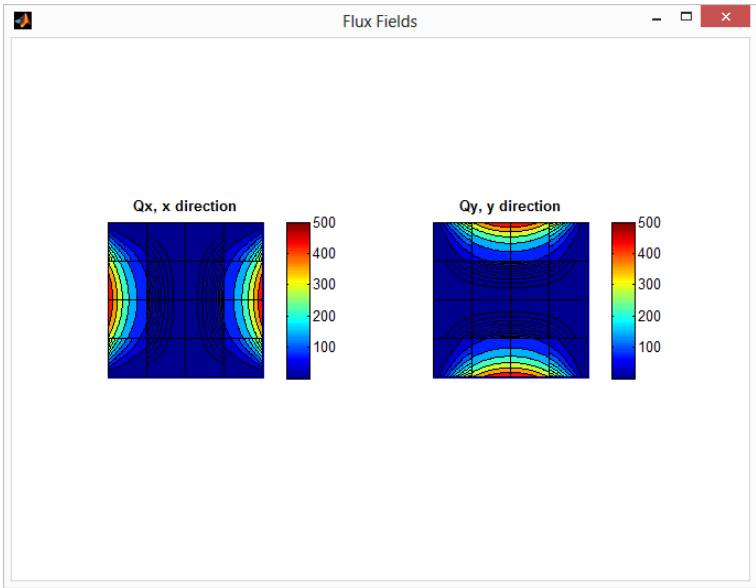


Figura B.18: Exemplo de campo de fluxos

De seguida, os graus resultantes do refinamento, utilizados na solução final são apresentados dispostos na malha dos elementos. Os graus das bases no domínio e nas fronteiras, são representados a azul e verde, respectivamente.

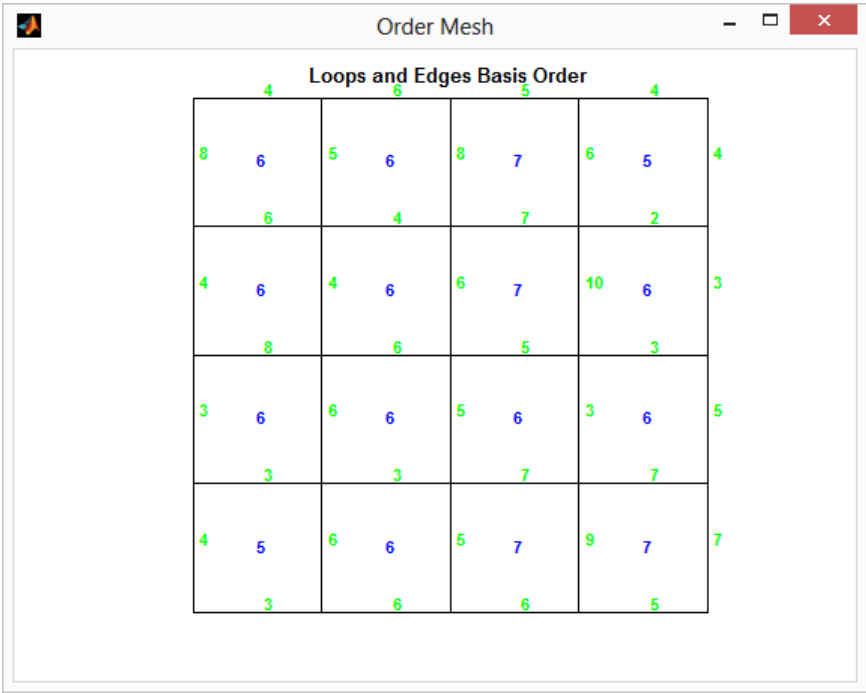


Figura B.19: Graus da malha resultante do refinamento.

Finalmente, as duas últimas janelas apresentam o progresso do processo iterativo, detalhando todo o processo, iteração a iteração, e fornecendo diferentes informações relevantes sobre a tomada de decisão. A primeira é uma tabela (Figura B.20) apresentando a indeterminação cinemática (β), quais os domínios e fronteiras refinadas em cada iteração, os graus dos domínios e fronteiras em cada iteração e ainda a energia do sistema após cada refinamento.

[illegible]

Figura B.20: Tabela com informações sobre cada iteração

A segunda janela (Figura B.21) refere-se a um conjunto de gráficos que permite avaliar a convergência e estabilidade do processo iterativo. São apresentados as variações de energia, a norma do seu erro, bem como as normas das densidades dos erros nos domínios e nas fronteiras.

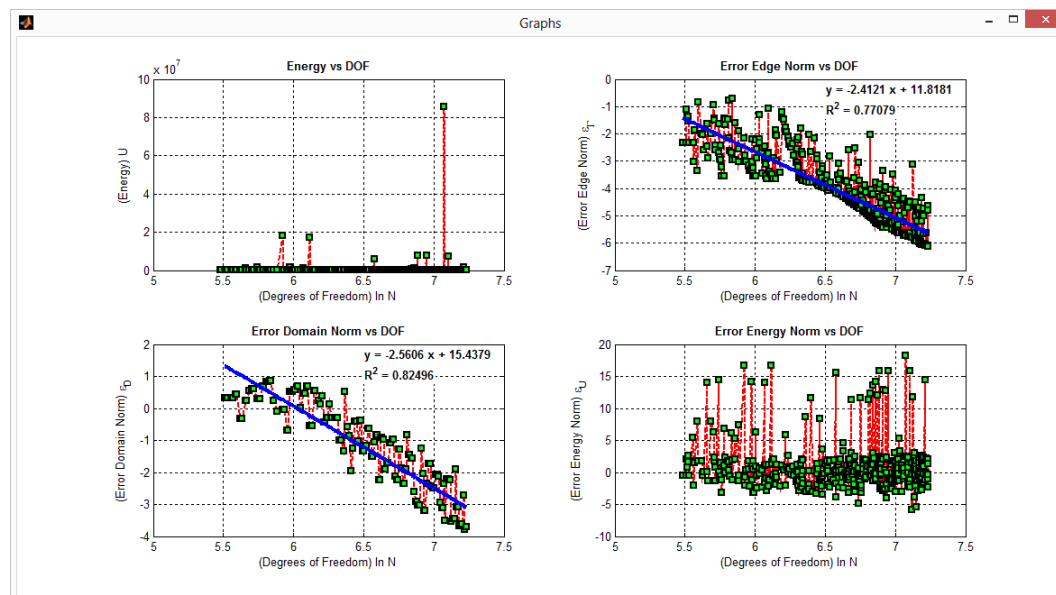


Figura B.21: Gráficos finais

Listagem das estruturas e sub-estruturas utilizadas

`Loops.nodes` : **matriz**, de dimensão do número total de elementos por quatro, em que cada linha contém os nós de cada elemento, numa ordem em contra-relógio, começando no nó inferior esquerdo;

`Loops.edges` : **matriz**, de dimensão do número total de fronteiras por quatro, em que cada linha contém as quatro fronteiras pertencentes a esse elemento;

`Loops.center` : **matriz**, de dimensão do número total de elementos por dois, em que cada linha contém as coordenadas cartesianas globais, correspondendo à origem do referencial local do elemento. A origem do referencial global é situada, por defeito, no nó inferior esquerdo da malha;

`Loops.area` : **vector**, de dimensão do número total de elementos, em que cada linha é a área do elemento respectivo;

`Loops.order` : **vector**, de dimensão do número total de elementos, em que cada linha corresponde ao grau da base de funções de aproximação do domínio respectivo;

`Loops.dim` : **vector**, com a dimensão do número total de elementos, em que cada linha é a dimensão de cada submatriz (quadrada) de condutividade do respectivo elemento;

`Loops.insert` : **vector**, com a dimensão do número total de elementos, em que em cada linha está a *posição de entrada* de cada submatriz de condutividade no sistema global;

`Edges.nini` : **vector**, de dimensão do número total de fronteiras, com os nós iniciais de cada fronteira;

`Edges.nfin` : **vector**, de dimensão do número total de fronteiras, com os nós finais de cada fronteira;

`Edges.parametric` : **matriz**, de dimensão do número total de fronteiras por quatro, onde cada linha indica as coordenadas cartesianas globais do nó inicial e a variação em coordenadas cartesianas em x e y da fronteira;

`Edges.lleft` : **vector**, de dimensão do número total de fronteiras, com o elemento à esquerda da fronteira;

`Edges.lright` : **vector**, de dimensão do número total de fronteiras, com o elemento à direita da fronteira;

`Edges.type` : **vector**, de dimensão do número total de fronteiras, com o tipo de fronteira para cada uma, ‘D’ para Dirichlet, ‘N’ para Neumann. As fronteiras interiores são definidas como Dirichlet;

`Edges.order` : **vector**, de dimensão do número total de fronteiras, contendo o grau da base das funções de aproximação para a fronteira respectiva;

`Edges.dim` : **vector**, de dimensão do número total de fronteiras, contendo a dimensão da base de aproximação nas fronteiras;

`Edges.insert` : **vector**, de dimensão do número total de fronteiras, com a *posição de entrada* de cada submatriz **B** no sistema global;

`BConds.Dirichlet` : **vector**, de dimensão do número total de fronteiras, onde os valores das temperaturas nas fronteiras de Dirichlet são armazenados nas posições correspondentes às respectivas fronteiras, e as restantes entradas são ‘NaN’ — operador sem input;

`BConds.Neumann` : **vector**, de dimensão do número total de fronteiras, com os valores dos fluxos nas fronteiras de Neumann que são armazenados nas posições correspondentes às respectivas fronteiras, e as restantes entradas são ‘NaN’.

Apêndice C

Tabelas do refinamento iterativo para os casos de estudo

Tabela C.1: Processo iterativo para o caso de estudo 1

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
1	60	4 4 4 4	None	3 3 3 3	7 2 10 1 9 6 12 5	1 1 0 0 1 1 1 0 1 1 0 1	1.8155e+03
2	68	2 2 2 2	None	3 3 3 3	6 9 1 2 10 5 12 7	2 2 0 0 2 2 2 0 2 2 0 2	1.7717e+03
3	96	0 0 0 0	1 2 3 4	4 4 4 4	5 7 9 6 1 2 12 10	3 3 0 0 3 3 3 0 3 3 0 3	1.7965e+03
4	128	3 3 3 3	1 2 3 4	5 5 5 5	9 1 6 12 2 5 10 7	4 4 0 0 4 4 4 0 4 4 0 4	1.7986e+03
5	164	7 7 7 7	1 2 3 4	6 6 6 6	6 12 1 5 9 7 2 10	5 5 0 0 5 5 5 0 5 5 0 5	1.7992e+03
6	204	12 12 12 12	1 2 3 4	7 7 7 7	1 6 7 5 12 10 2 9	6 6 0 0 6 6 6 0 6 6 0 6	1.7993e+03
7	248	18 18 18 18	1 2 3 4	8 8 8 8	2 12 10 6 1 7 5 9	7 7 0 0 7 7 7 0 7 7 0 7	1.7994e+03
8	296	25 25 25 25	1 2 3 4	9 9 9 9	12 6 10 5 2 9 7 1	8 8 0 0 8 8 8 0 8 8 0 8	1.7994e+03
9	348	33 33 33 33	1 2 3 4	10 10 10 10	2 9 10 5 6 12 1 7	9 9 0 0 9 9 9 0 9 9 0 9	1.7994e+03
10	404	42 42 42 42	1 2 3 4	11 11 11 11	7 1 6 12 10 5 9 2	10 10 0 0 10 10 10 0 10 10 0 10	1.7994e+03

Continua na página seguinte

Tabela C.1 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
11	464	52 52 52 52	1 2 3 4	12 12 12 12	10 2 6 12 9 5 1 7	11 11 0 0 11 11 11 0 11 11 0 11	1.7994e+03
12	528	63 63 63 63	1 2 3 4	13 13 13 13	9 2 10 12 7 6 5 1	12 12 0 0 12 12 12 0 12 12 0 12	1.7994e+03
13	596	75 75 75 75	1 2 3 4	14 14 14 14	9 7 12 1 10 6 5 2	13 13 0 0 13 13 13 0 13 13 0 13	1.7994e+03
14	668	88 88 88 88	1 2 3 4	15 15 15 15	9 7 12 1 10 6 5 2	14 14 0 0 14 14 14 0 14 14 0 14	1.7994e+03

Tabela C.2: Processo iterativo para o caso de estudo 2

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
1	57	6 4 4 1	None	3 3 3 3	12	ND ND 1 1 1 1 ND 1 1 ND 1 2	4.9968e+03
2	63	6 4 4 0	4	3 3 3 4	6	ND ND 1 1 1 2 ND 1 1 ND 1 2	5.0028e+03
3	64	6 3 4 5	None	3 3 3 4	9	ND ND 1 1 1 2 ND 1 2 ND 1 2	5.0013e+03
4	65	6 3 4 4	None	3 3 3 4	6	ND ND 1 1 1 3 ND 1 2 ND 1 2	5.0006e+03
5	66	6 3 3 3	None	3 3 3 4	11	ND ND 1 1 1 3 ND 1 2 ND 2 2	5.0003e+03
6	67	6 3 2 2	None	3 3 3 4	11	ND ND 1 1 1 3 ND 1 2 ND 3 2	5.0002e+03

Continua na página seguinte

Tabela C.2 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
7	68	6 3 1 1	None	3 3 3 4	11	ND ND 1 1 1 3 ND 1 2 ND 4 2	5.0002e+03
8	75	6 3 1 0	4	3 3 3 5	6	ND ND 1 1 1 4 ND 1 2 ND 4 2	5.0007e+03
9	76	6 3 1 5	None	3 3 3 5	12	ND ND 1 1 1 4 ND 1 2 ND 4 3	5.0007e+03
10	77	6 3 1 4	None	3 3 3 5	12	ND ND 1 1 1 4 ND 1 2 ND 4 4	5.0007e+03
11	78	6 3 1 3	None	3 3 3 5	12	ND ND 1 1 1 4 ND 1 2 ND 4 5	5.0007e+03
12	79	6 3 1 2	None	3 3 3 5	6	ND ND 1 1 1 5 ND 1 2 ND 4 5	5.0007e+03
13	80	6 3 1 1	None	3 3 3 5	6	ND ND 1 1 1 6 ND 1 2 ND 4 5	5.0007e+03
14	88	6 3 1 0	4	3 3 3 6	12	ND ND 1 1 1 6 ND 1 2 ND 4 6	5.0008e+03
15	89	6 2 1 6	None	3 3 3 6	4	ND ND 1 2 1 6 ND 1 2 ND 4 6	5.0008e+03
16	95	6 2 0 5	3	3 3 4 6	11	ND ND 1 2 1 6 ND 1 2 ND 5 6	5.0009e+03
17	96	5 2 4 5	None	3 3 4 6	3	ND ND 2 2 1 6 ND 1 2 ND 5 6	5.0008e+03
18	97	4 1 4 5	None	3 3 4 6	8	ND ND 2 2 1 6 ND 2 2 ND 5 6	5.0007e+03

Continua na página seguinte

Tabela C.2 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
19	103	4 0 4 4	2	3 4 4 6	4	ND ND 2 3 1 6 ND 2 2 ND 5 6	5.0019e+03
20	104	3 4 4 4	None	3 4 4 6	8	ND ND 2 3 1 6 ND 3 2 ND 5 6	5.0019e+03
21	105	2 3 4 4	None	3 4 4 6	8	ND ND 2 3 1 6 ND 4 2 ND 5 6	5.0018e+03
22	106	2 2 4 3	None	3 4 4 6	4	ND ND 2 4 1 6 ND 4 2 ND 5 6	5.0017e+03
23	107	2 1 4 3	None	3 4 4 6	9	ND ND 2 4 1 6 ND 4 3 ND 5 6	5.0017e+03
24	114	2 0 4 3	2	3 5 4 6	9	ND ND 2 4 1 6 ND 4 4 ND 5 6	5.0019e+03
25	115	2 6 3 3	None	3 5 4 6	5	ND ND 2 4 2 6 ND 4 4 ND 5 6	5.0019e+03
26	116	2 5 3 3	None	3 5 4 6	9	ND ND 2 4 2 6 ND 4 5 ND 5 6	5.0019e+03
27	117	2 4 3 3	None	3 5 4 6	9	ND ND 2 4 2 6 ND 4 6 ND 5 6	5.0019e+03
28	118	2 3 3 2	None	3 5 4 6	4	ND ND 2 5 2 6 ND 4 6 ND 5 6	5.0019e+03
29	119	2 3 3 1	None	3 5 4 6	6	ND ND 2 5 2 7 ND 4 6 ND 5 6	5.0019e+03

Tabela C.3: Processo iterativo para o caso de estudo 3

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
1	242	2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1	None	3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	25 40	1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2	2.1263e+03
2	254	2 2 2 0 2 2 2 2 2 2 2 2 2 2 2 0	4 16	3 3 3 4 3 3 3 3 3 3 3 3 3 3 3 4	20 4	1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2	1.9995e+03
3	256	2 2 2 4 2 2 2 1 2 2 2 1 2 2 2 4	None	3 3 3 4 3 3 3 3 3 3 3 3 3 3 3 4	8 16	1 1 1 2 1 1 1 2 1 1 1 1 1 1 1 2 1 1 1 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2	1.9995e+03
4	258	2 2 1 3 2 2 2 1 2 2 2 1 2 2 1 3	None	3 3 3 4 3 3 3 3 3 3 3 3 3 3 3 4	24 39	1 1 1 2 1 1 1 2 1 1 1 1 1 1 1 2 1 1 1 2 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2	2.0047e+03
5	270	2 2 0 2 2 2 2 1 2 2 2 1 2 2 0 2	3 15	3 3 4 4 3 3 3 3 3 3 3 3 3 3 4 4	24 39	1 1 1 2 1 1 1 2 1 1 1 1 1 1 1 2 1 1 1 2 1 1 1 3 2 1 1 1 1 1 1 1 1 1 1 1 1 1 3 2	1.9590e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
11	314	2 2 2 0 2 2 5 3 2 2 5 3 2 2 2 0	4 16	3 3 4 5 3 3 4 4 3 3 4 4 3 3 4 5	39 24	1 1 2 2 1 1 2 3 1 1 1 1 1 1 2 3 1 1 2 2 1 1 1 4 2 1 1 1 2 2 1 1 1 2 2 1 1 1 4 2	1.4636e+03
12	316	2 2 2 5 2 2 5 3 2 2 5 3 2 2 2 5	None	3 3 4 5 3 3 4 4 3 3 4 4 3 3 4 5	20 4	1 1 2 3 1 1 2 3 1 1 1 1 1 1 2 3 1 1 2 3 1 1 1 4 2 1 1 1 2 2 1 1 1 2 2 1 1 1 4 2	1.7325e+03
13	318	2 2 2 4 2 2 5 3 2 2 5 3 2 2 2 4	None	3 3 4 5 3 3 4 4 3 3 4 4 3 3 4 5	40 25	1 1 2 3 1 1 2 3 1 1 1 1 1 1 2 3 1 1 2 3 1 1 1 4 3 1 1 1 2 2 1 1 1 2 2 1 1 1 4 3	2.8712e+03
14	320	2 2 1 4 2 2 5 3 2 2 5 3 2 2 1 4	None	3 3 4 5 3 3 4 4 3 3 4 4 3 3 4 5	19 3	1 1 3 3 1 1 2 3 1 1 1 1 1 1 2 3 1 1 3 3 1 1 1 4 3 1 1 1 2 2 1 1 1 2 2 1 1 1 4 3	3.4709e+03
15	334	2 1 0 4 2 2 5 3 2 2 5 3 2 1 0 4	3 15	3 3 5 5 3 3 4 4 3 3 4 4 3 3 5 5	23 38	1 1 3 3 1 1 2 3 1 1 1 1 1 1 2 3 1 1 3 3 1 1 2 4 3 1 1 1 2 2 1 1 1 2 2 1 1 2 4 3	2.8101e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
16	346	2 0 5 4 2 2 5 3 2 2 5 3 2 0 5 4	2 14	3 4 5 5 3 3 4 4 3 3 4 4 3 4 5 5	38 23	1 1 3 3 1 1 2 3 1 1 1 1 1 1 2 3 1 1 3 3 1 1 3 4 3 1 1 1 2 2 1 1 1 2 2 1 1 3 4 3	2.8384e+03
17	348	2 5 5 3 2 2 5 2 2 2 5 2 2 5 5 3	None	3 4 5 5 3 3 4 4 3 3 4 4 3 4 5 5	8 16	1 1 3 3 1 1 2 4 1 1 1 1 1 1 2 4 1 1 3 3 1 1 3 4 3 1 1 1 2 2 1 1 1 2 2 1 1 3 4 3	3.1449e+03
18	350	2 5 5 3 2 2 5 1 2 2 5 1 2 5 5 3	None	3 4 5 5 3 3 4 4 3 3 4 4 3 4 5 5	30 35	1 1 3 3 1 1 2 4 1 1 1 1 1 1 2 4 1 1 3 3 1 1 3 4 3 1 1 1 2 3 1 1 1 2 3 1 1 3 4 3	3.8272e+03
19	352	2 5 4 3 2 2 5 1 2 2 5 1 2 5 4 3	None	3 4 5 5 3 3 4 4 3 3 4 4 3 4 5 5	3 19	1 1 4 3 1 1 2 4 1 1 1 1 1 1 2 4 1 1 4 3 1 1 3 4 3 1 1 1 2 3 1 1 1 2 3 1 1 3 4 3	4.6719e+03
20	354	2 5 3 3 2 2 5 1 2 2 5 1 2 5 3 3	None	3 4 5 5 3 3 4 4 3 3 4 4 3 4 5 5	19 3	1 1 5 3 1 1 2 4 1 1 1 1 1 1 2 4 1 1 5 3 1 1 3 4 3 1 1 1 2 3 1 1 1 2 3 1 1 3 4 3	5.2358e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
21	368	2 5 3 2 2 2 5 0 2 2 5 0 2 5 3 2	8 12	3 4 5 5 3 3 4 5 3 3 4 5 3 4 5 5	8 16	1 1 5 3 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 3 1 1 3 4 3 1 1 1 2 3 1 1 1 2 3 1 1 3 4 3	3.9234e+03
22	370	2 5 3 1 2 2 5 6 2 2 5 6 2 5 3 1	None	3 4 5 5 3 3 4 5 3 3 4 5 3 4 5 5	25 40	1 1 5 3 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 3 1 1 3 4 4 1 1 1 2 3 1 1 1 2 3 1 1 3 4 4	7.2978e+03
23	372	2 5 3 1 2 2 5 5 2 2 5 5 2 5 3 1	None	3 4 5 5 3 3 4 5 3 3 4 5 3 4 5 5	35 30	1 1 5 3 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 3 1 1 3 4 4 1 1 1 2 4 1 1 1 2 4 1 1 3 4 4	7.4049e+03
24	388	2 5 3 0 2 2 5 5 2 2 5 5 2 5 3 0	4 16	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	20 4	1 1 5 4 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 4 1 1 3 4 4 1 1 1 2 4 1 1 1 2 4 1 1 3 4 4	3.3109e+03
25	390	2 5 2 6 2 2 5 5 2 2 5 5 2 5 2 6	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	24 39	1 1 5 4 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 4 1 1 3 5 4 1 1 1 2 4 1 1 1 2 4 1 1 3 5 4	3.3121e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
26	392	2 5 2 5 2 2 5 5 2 2 5 5 2 5 2 5	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	40 25	1 1 5 4 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 4 1 1 3 5 5 1 1 1 2 4 1 1 1 2 4 1 1 3 5 5	8.7699e+03
27	394	2 5 2 5 2 2 5 4 2 2 5 4 2 5 2 5	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	30 35	1 1 5 4 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 4 1 1 3 5 5 1 1 1 2 5 1 1 1 2 5 1 1 3 5 5	3.6858e+03
28	396	2 5 2 4 2 2 5 4 2 2 5 4 2 5 2 4	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	4 20	1 1 5 5 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 5 1 1 3 5 5 1 1 1 2 5 1 1 1 2 5 1 1 3 5 5	3.2201e+03
29	398	2 5 2 4 2 2 4 3 2 2 4 3 2 5 2 4	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	34 29	1 1 5 5 1 1 2 5 1 1 1 1 1 1 2 5 1 1 5 5 1 1 3 5 5 1 1 1 3 5 1 1 1 3 5 1 1 3 5 5	3.2663e+03
30	400	2 5 1 4 2 2 3 3 2 2 3 3 2 5 1 4	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	15 7	1 1 5 5 1 1 3 5 1 1 1 1 1 1 3 5 1 1 5 5 1 1 3 5 5 1 1 1 3 5 1 1 1 3 5 1 1 3 5 5	3.4773e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
31	402	2 5 1 4 2 1 2 3 2 1 2 3 2 5 1 4	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	28 33	1 1 5 5 1 1 3 5 1 1 1 1 1 1 3 5 1 1 5 5 1 1 3 5 5 1 1 2 3 5 1 1 2 3 5 1 1 3 5 5	3.4962e+03
32	404	2 5 1 4 2 1 1 2 2 1 1 2 2 5 1 4	None	3 4 5 6 3 3 4 5 3 3 4 5 3 4 5 6	29 34	1 1 5 5 1 1 3 5 1 1 1 1 1 1 3 5 1 1 5 5 1 1 3 5 5 1 1 2 4 5 1 1 2 4 5 1 1 3 5 5	3.8349e+03
33	432	2 5 0 4 2 1 0 2 2 1 0 2 2 5 0 4	3 7 11 15	3 4 6 6 3 3 5 5 3 3 5 5 3 4 6 6	15 7	1 1 5 5 1 1 4 5 1 1 1 1 1 1 4 5 1 1 5 5 1 1 3 5 5 1 1 2 4 5 1 1 2 4 5 1 1 3 5 5	7.2835e+03
34	434	2 4 6 4 2 1 6 2 2 1 6 2 2 4 6 4	None	3 4 6 6 3 3 5 5 3 3 5 5 3 4 6 6	23 38	1 1 5 5 1 1 4 5 1 1 1 1 1 1 4 5 1 1 5 5 1 1 4 5 5 1 1 2 4 5 1 1 2 4 5 1 1 4 5 5	7.6544e+03
35	436	2 3 6 4 2 1 6 2 2 1 6 2 2 3 6 4	None	3 4 6 6 3 3 5 5 3 3 5 5 3 4 6 6	18 2	1 2 5 5 1 1 4 5 1 1 1 1 1 1 4 5 1 2 5 5 1 1 4 5 5 1 1 2 4 5 1 1 2 4 5 1 1 4 5 5	8.1003e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
36	448	2 3 6 4 2 0 5 2 2 0 5 2 2 3 6 4	6 10	3 4 6 6 3 4 5 5 3 4 5 5 3 4 6 6	28 33	1 2 5 5 1 1 4 5 1 1 1 1 1 1 4 5 1 2 5 5 1 1 4 5 5 1 1 3 4 5 1 1 3 4 5 1 1 4 5 5	8.0954e+03
37	450	2 2 6 4 2 5 5 2 2 5 5 2 2 2 6 4	None	3 4 6 6 3 4 5 5 3 4 5 5 3 4 6 6	18 2	1 3 5 5 1 1 4 5 1 1 1 1 1 1 4 5 1 3 5 5 1 1 4 5 5 1 1 3 4 5 1 1 3 4 5 1 1 4 5 5	8.6613e+03
38	452	1 1 6 4 2 5 5 2 2 5 5 2 1 1 6 4	None	3 4 6 6 3 4 5 5 3 4 5 5 3 4 6 6	22 37	1 3 5 5 1 1 4 5 1 1 1 1 1 1 4 5 1 3 5 5 1 2 4 5 5 1 1 3 4 5 1 1 3 4 5 1 2 4 5 5	8.6861e+03
39	466	1 0 6 4 2 5 5 2 2 5 5 2 1 0 6 4	2 14	3 5 6 6 3 4 5 5 3 4 5 5 3 5 6 6	18 2	1 4 5 5 1 1 4 5 1 1 1 1 1 1 4 5 1 4 5 5 1 2 4 5 5 1 1 3 4 5 1 1 3 4 5 1 2 4 5 5	8.4510e+03
40	468	1 5 6 4 2 4 5 2 2 4 5 2 1 5 6 4	None	3 5 6 6 3 4 5 5 3 4 5 5 3 5 6 6	6 14	1 4 5 5 1 2 4 5 1 1 1 1 1 2 4 5 1 4 5 5 1 2 4 5 5 1 1 3 4 5 1 1 3 4 5 1 2 4 5 5	8.6780e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
41	470	1 4 6 4 2 3 5 2 2 3 5 2 1 4 6 4	None	3 5 6 6 3 4 5 5 3 4 5 5 3 5 6 6	6 14	1 4 5 5 1 3 4 5 1 1 1 1 1 3 4 5 1 4 5 5 1 2 4 5 5 1 1 3 4 5 1 1 3 4 5 1 2 4 5 5	8.9236e+03
42	472	1 3 6 4 2 2 5 2 2 2 5 2 1 3 6 4	None	3 5 6 6 3 4 5 5 3 4 5 5 3 5 6 6	6 14	1 4 5 5 1 4 4 5 1 1 1 1 1 4 4 5 1 4 5 5 1 2 4 5 5 1 1 3 4 5 1 1 3 4 5 1 2 4 5 5	9.2129e+03
43	474	1 3 6 4 2 1 4 2 2 1 4 2 1 3 6 4	None	3 5 6 6 3 4 5 5 3 4 5 5 3 5 6 6	28 33	1 4 5 5 1 4 4 5 1 1 1 1 1 4 4 5 1 4 5 5 1 2 4 5 5 1 1 4 4 5 1 1 4 4 5 1 2 4 5 5	1.0027e+04
44	476	1 2 6 4 2 1 4 2 2 1 4 2 1 2 6 4	None	3 5 6 6 3 4 5 5 3 4 5 5 3 5 6 6	2 18	1 5 5 5 1 4 4 5 1 1 1 1 1 4 4 5 1 5 5 5 1 2 4 5 5 1 1 4 4 5 1 1 4 4 5 1 2 4 5 5	1.3772e+04
45	490	1 2 6 4 2 0 3 2 2 0 3 2 1 2 6 4	6 10	3 5 6 6 3 5 5 5 3 5 5 5 3 5 6 6	33 28	1 5 5 5 1 4 4 5 1 1 1 1 1 4 4 5 1 5 5 5 1 2 4 5 5 1 1 5 4 5 1 1 5 4 5 1 2 4 5 5	1.9326e+04

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
46	492	1 2 6 4 1 5 3 2 1 5 3 2 1 2 6 4	None	3 5 6 6 3 5 5 5 3 5 5 5 3 5 6 6	27 32	1 5 5 5 1 4 4 5 1 1 1 1 1 4 4 5 1 5 5 5 1 2 4 5 5 1 2 5 4 5 1 2 5 4 5 1 2 4 5 5	1.9416e+04
47	494	1 1 6 4 1 4 3 2 1 4 3 2 1 1 6 4	None	3 5 6 6 3 5 5 5 3 5 5 5 3 5 6 6	14 6	1 5 5 5 1 5 4 5 1 1 1 1 1 5 4 5 1 5 5 5 1 2 4 5 5 1 2 5 4 5 1 2 5 4 5 1 2 4 5 5	4.3342e+04
48	496	1 1 6 4 1 4 2 1 1 4 2 1 1 1 6 4	None	3 5 6 6 3 5 5 5 3 5 5 5 3 5 6 6	29 34	1 5 5 5 1 5 4 5 1 1 1 1 1 5 4 5 1 5 5 5 1 2 4 5 5 1 2 5 5 5 1 2 5 5 5 1 2 4 5 5	5.6920e+03
49	498	1 1 5 4 1 4 1 1 1 4 1 1 1 1 5 4	None	3 5 6 6 3 5 5 5 3 5 5 5 3 5 6 6	7 15	1 5 5 5 1 5 5 5 1 1 1 1 1 5 5 5 1 5 5 5 1 2 4 5 5 1 2 5 5 5 1 2 5 5 5 1 2 4 5 5	3.5918e+03
50	500	1 1 5 3 1 4 1 1 1 4 1 1 1 1 5 3	None	3 5 6 6 3 5 5 5 3 5 5 5 3 5 6 6	4 20	1 5 5 6 1 5 5 5 1 1 1 1 1 5 5 5 1 5 5 6 1 2 4 5 5 1 2 5 5 5 1 2 5 5 5 1 2 4 5 5	4.9617e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
51	502	1 1 5 2 1 4 1 1 1 4 1 1 1 1 5 2	None	3 5 6 6 3 5 5 5 3 5 5 5 3 5 6 6	40 25	1 5 5 6 1 5 5 5 1 1 1 1 1 5 5 5 1 5 5 6 1 2 4 5 6 1 2 5 5 5 1 2 5 5 5 1 2 4 5 6	1.7502e+03
52	518	1 0 4 2 1 4 1 1 1 4 1 1 1 0 4 2	2 14	3 6 6 6 3 5 5 5 3 5 5 5 3 6 6 6	38 23	1 5 5 6 1 5 5 5 1 1 1 1 1 5 5 5 1 5 5 6 1 2 5 5 6 1 2 5 5 5 1 2 5 5 5 1 2 5 5 6	2.5159e+03
53	530	1 7 4 2 0 3 1 1 0 3 1 1 1 7 4 2	5 9	3 6 6 6 4 5 5 5 4 5 5 5 3 6 6 6	32 27	1 5 5 6 1 5 5 5 1 1 1 1 1 5 5 5 1 5 5 6 1 2 5 5 6 1 3 5 5 5 1 3 5 5 5 1 2 5 5 6	2.5156e+03
54	542	0 6 4 2 5 3 1 1 5 3 1 1 0 6 4 2	1 13	4 6 6 6 4 5 5 5 4 5 5 5 4 6 6 6	37 22	1 5 5 6 1 5 5 5 1 1 1 1 1 5 5 5 1 5 5 6 1 3 5 5 6 1 3 5 5 5 1 3 5 5 5 1 3 5 5 6	2.5141e+03
55	544	4 5 4 2 5 3 1 1 5 3 1 1 4 5 4 2	None	4 6 6 6 4 5 5 5 4 5 5 5 4 6 6 6	37 22	1 5 5 6 1 5 5 5 1 1 1 1 1 5 5 5 1 5 5 6 1 4 5 5 6 1 3 5 5 5 1 3 5 5 5 1 4 5 5 6	2.5703e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
61	568	4 5 4 2 3 3 1 1 3 3 1 1 4 5 4 2	None	5 6 6 6 4 5 5 5 4 5 5 5 5 6 6 6	5 13	4 5 5 6 3 5 5 5 1 1 1 1 3 5 5 5 4 5 5 6 2 4 5 5 6 1 3 5 5 5 1 3 5 5 5 2 4 5 5 6	2.7670e+03
62	570	3 5 4 2 2 3 1 1 2 3 1 1 3 5 4 2	None	5 6 6 6 4 5 5 5 4 5 5 5 5 6 6 6	13 5	4 5 5 6 4 5 5 5 1 1 1 1 4 5 5 5 4 5 5 6 2 4 5 5 6 1 3 5 5 5 1 3 5 5 5 2 4 5 5 6	2.8404e+03
63	572	3 5 4 2 1 2 1 1 1 2 1 1 3 5 4 2	None	5 6 6 6 4 5 5 5 4 5 5 5 5 6 6 6	32 27	4 5 5 6 4 5 5 5 1 1 1 1 4 5 5 5 4 5 5 6 2 4 5 5 6 1 4 5 5 5 1 4 5 5 5 2 4 5 5 6	3.1946e+03
64	574	2 5 4 2 1 2 1 1 1 2 1 1 2 5 4 2	None	5 6 6 6 4 5 5 5 4 5 5 5 5 6 6 6	1 17	5 5 5 6 4 5 5 5 1 1 1 1 4 5 5 5 5 5 5 6 2 4 5 5 6 1 4 5 5 5 1 4 5 5 5 2 4 5 5 6	1.9034e+03
65	588	2 5 4 2 0 1 1 1 0 1 1 1 2 5 4 2	5 9	5 6 6 6 5 5 5 5 5 5 5 5 5 6 6 6	32 27	5 5 5 6 4 5 5 5 1 1 1 1 4 5 5 5 5 5 5 6 2 4 5 5 6 1 5 5 5 5 1 5 5 5 5 2 4 5 5 6	5.2073e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
71	616	1 2 2 1 3 6 1 1 3 6 1 1 1 2 2 1	None	5 6 6 6 5 6 5 5 5 6 5 5 5 6 6 6	32 27	5 6 5 6 5 6 5 5 1 1 1 1 5 6 5 5 5 6 5 6 2 4 6 6 6 2 6 5 5 5 2 6 5 5 5 2 4 6 6 6	2.0922e+03
72	618	1 2 1 1 3 6 1 1 3 6 1 1 1 2 1 1	None	5 6 6 6 5 6 5 5 5 6 5 5 5 6 6 6	3 19	5 6 6 6 5 6 5 5 1 1 1 1 5 6 5 5 5 6 6 6 2 4 6 6 6 2 6 5 5 5 2 6 5 5 5 2 4 6 6 6	2.0741e+03
73	650	1 2 0 1 3 6 0 1 3 6 0 1 1 2 0 1	3 7 11 15	5 6 7 6 5 6 6 5 5 6 6 5 5 6 7 6	7 15	5 6 6 6 5 6 6 5 1 1 1 1 5 6 6 5 5 6 6 6 2 4 6 6 6 2 6 5 5 5 2 6 5 5 5 2 4 6 6 6	2.0529e+03
74	652	1 2 8 1 3 5 6 1 3 5 6 1 1 2 8 1	None	5 6 7 6 5 6 6 5 5 6 6 5 5 6 7 6	28 33	5 6 6 6 5 6 6 5 1 1 1 1 5 6 6 5 5 6 6 6 2 4 6 6 6 2 6 6 5 5 2 6 6 5 5 2 4 6 6 6	1.8504e+03
75	654	1 2 7 1 3 5 6 1 3 5 6 1 1 2 7 1	None	5 6 7 6 5 6 6 5 5 6 6 5 5 6 7 6	19 3	5 6 7 6 5 6 6 5 1 1 1 1 5 6 6 5 5 6 7 6 2 4 6 6 6 2 6 6 5 5 2 6 6 5 5 2 4 6 6 6	1.7925e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
81	696	1 1 4 6 3 5 4 7 3 5 4 7 1 1 4 6	None	5 6 7 7 5 6 6 6 5 6 6 6 5 6 7 7	4 20	5 6 7 7 5 6 8 6 1 1 1 1 5 6 8 6 5 6 7 7 2 4 7 6 7 2 6 6 5 5 2 6 6 5 5 2 4 7 6 7	2.1267e+03
82	698	1 1 3 5 3 5 4 7 3 5 4 7 1 1 3 5	None	5 6 7 7 5 6 6 6 5 6 6 6 5 6 7 7	24 39	5 6 7 7 5 6 8 6 1 1 1 1 5 6 8 6 5 6 7 7 2 4 7 7 7 2 6 6 5 5 2 6 6 5 5 2 4 7 7 7	1.8626e+03
83	700	1 1 3 4 3 5 4 6 3 5 4 6 1 1 3 4	None	5 6 7 7 5 6 6 6 5 6 6 6 5 6 7 7	16 8	5 6 7 7 5 6 8 7 1 1 1 1 5 6 8 7 5 6 7 7 2 4 7 7 7 2 6 6 5 5 2 6 6 5 5 2 4 7 7 7	1.7152e+03
84	702	1 1 3 3 3 5 4 5 3 5 4 5 1 1 3 3	None	5 6 7 7 5 6 6 6 5 6 6 6 5 6 7 7	8 16	5 6 7 7 5 6 8 8 1 1 1 1 5 6 8 8 5 6 7 7 2 4 7 7 7 2 6 6 5 5 2 6 6 5 5 2 4 7 7 7	1.7152e+03
85	704	1 1 3 2 3 5 4 5 3 5 4 5 1 1 3 2	None	5 6 7 7 5 6 6 6 5 6 6 6 5 6 7 7	20 4	5 6 7 8 5 6 8 8 1 1 1 1 5 6 8 8 5 6 7 8 2 4 7 7 7 2 6 6 5 5 2 6 6 5 5 2 4 7 7 7	1.7789e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
86	706	1 1 2 1 3 5 4 5 3 5 4 5 1 1 2 1	None	5 6 7 7 5 6 6 6 5 6 6 6 5 6 7 7	24 39	5 6 7 8 5 6 8 8 1 1 1 1 5 6 8 8 5 6 7 8 2 4 7 8 7 2 6 6 5 5 2 6 6 5 5 2 4 7 8 7	1.7858e+03
87	728	1 1 2 -1 3 5 4 4 3 5 4 4 1 1 2 -1	4 16	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	16 8 20 4	5 6 7 9 5 6 8 9 1 1 1 1 5 6 8 9 5 6 7 9 2 4 7 8 7 2 6 6 5 5 2 6 6 5 5 2 4 7 8 7	1.8823e+03
88	730	1 1 2 8 3 5 3 3 3 5 3 3 1 1 2 8	None	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	34 29	5 6 7 9 5 6 8 9 1 1 1 1 5 6 8 9 5 6 7 9 2 4 7 8 7 2 6 6 6 5 2 6 6 6 5 2 4 7 8 7	1.5475e+03
89	732	1 1 1 7 3 5 3 3 3 5 3 3 1 1 1 7	None	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	39 24	5 6 7 9 5 6 8 9 1 1 1 1 5 6 8 9 5 6 7 9 2 4 7 9 7 2 6 6 6 5 2 6 6 6 5 2 4 7 9 7	1.5467e+03
90	734	1 1 1 6 3 5 3 3 3 5 3 3 1 1 1 6	None	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	4 20	5 6 7 10 5 6 8 9 1 1 1 1 5 6 8 9 5 6 7 10 2 4 7 9 7 2 6 6 6 5 2 6 6 6 5 2 4 7 9 7	1.5343e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
91	736	1 1 1 5 3 5 3 2 3 5 3 2 1 1 1 5	None	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	16 8	5 6 7 10 5 6 8 10 1 1 1 1 5 6 8 10 5 6 7 10 2 4 7 9 7 2 6 6 6 5 2 6 6 6 5 2 4 7 9 7	1.5295e+03
92	738	1 1 1 4 3 5 3 2 3 5 3 2 1 1 1 4	None	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	25 40	5 6 7 10 5 6 8 10 1 1 1 1 5 6 8 10 5 6 7 10 2 4 7 9 8 2 6 6 6 5 2 6 6 6 5 2 4 7 9 8	1.5699e+03
93	740	1 1 1 3 3 5 3 2 3 5 3 2 1 1 1 3	None	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	25 40	5 6 7 10 5 6 8 10 1 1 1 1 5 6 8 10 5 6 7 10 2 4 7 9 9 2 6 6 6 5 2 6 6 6 5 2 4 7 9 9	1.5352e+03
94	742	1 1 1 2 3 5 3 2 3 5 3 2 1 1 1 2	None	5 6 7 8 5 6 6 6 5 6 6 6 5 6 7 8	25 40	5 6 7 10 5 6 8 10 1 1 1 1 5 6 8 10 5 6 7 10 2 4 7 9 10 2 6 6 6 5 2 6 6 6 5 2 4 7 9 10	1.4875e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
95	762	1 1 0 1 3 5 3 2 3 5 3 2 1 1 0 1	3 15	5 6 8 8 5 6 6 6 5 6 6 6 5 6 8 8	24 39	5 6 7 10 5 6 8 10 1 1 1 1 5 6 8 10 5 6 7 10 2 4 7 10 10 2 6 6 6 5 2 6 6 6 5 2 4 7 10 10	1.7425e+03
96	764	1 1 8 1 3 5 3 2 3 5 3 2 1 1 8 1	None	5 6 8 8 5 6 6 6 5 6 6 6 5 6 8 8	19 3	5 6 8 10 5 6 8 10 1 1 1 1 5 6 8 10 5 6 8 10 2 4 7 10 10 2 6 6 6 5 2 6 6 6 5 2 4 7 10 10	1.7434e+03
97	782	1 0 7 1 3 5 3 2 3 5 3 2 1 0 7 1	2 14	5 7 8 8 5 6 6 6 5 6 6 6 5 7 8 8	38 23	5 6 8 10 5 6 8 10 1 1 1 1 5 6 8 10 5 6 8 10 2 4 8 10 10 2 6 6 6 5 2 6 6 6 5 2 4 8 10 10	1.7800e+03
98	784	1 7 7 1 3 5 3 2 3 5 3 2 1 7 7 1	None	5 7 8 8 5 6 6 6 5 6 6 6 5 7 8 8	2 18	5 7 8 10 5 6 8 10 1 1 1 1 5 6 8 10 5 7 8 10 2 4 8 10 10 2 6 6 6 5 2 6 6 6 5 2 4 8 10 10	1.5531e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
99	786	1 6 7 1 3 4 3 2 3 4 3 2 1 6 7 1	None	5 7 8 8 5 6 6 6 5 6 6 6 5 7 8 8	14 6	5 7 8 10 5 7 8 10 1 1 1 1 5 7 8 10 5 7 8 10 2 4 8 10 10 2 6 6 6 5 2 6 6 6 5 2 4 8 10 10	1.5513e+03
100	808	1 6 6 0 3 4 3 2 3 4 3 2 1 6 6 0	4 16	5 7 8 9 5 6 6 6 5 6 6 6 5 7 8 9	39 24	5 7 8 10 5 7 8 10 1 1 1 1 5 7 8 10 5 7 8 10 2 4 8 11 10 2 6 6 6 5 2 6 6 6 5 2 4 8 11 10	1.6221e+03
101	810	1 6 6 9 3 4 3 1 3 4 3 1 1 6 6 9	None	5 7 8 9 5 6 6 6 5 6 6 6 5 7 8 9	8 16	5 7 8 10 5 7 8 11 1 1 1 1 5 7 8 11 5 7 8 10 2 4 8 11 10 2 6 6 6 5 2 6 6 6 5 2 4 8 11 10	1.5808e+03
102	814	1 6 6 7 3 4 3 1 3 4 3 1 1 6 6 7	None	5 7 8 9 5 6 6 6 5 6 6 6 5 7 8 9	20 4 40 25	5 7 8 11 5 7 8 11 1 1 1 1 5 7 8 11 5 7 8 11 2 4 8 11 11 2 6 6 6 5 2 6 6 6 5 2 4 8 11 11	1.5076e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
103	818	1 6 6 5 3 4 3 1 3 4 3 1 1 6 6 5	None	5 7 8 9 5 6 6 6 5 6 6 6 5 7 8 9	40 25 20 4	5 7 8 12 5 7 8 11 1 1 1 1 5 7 8 11 5 7 8 12 2 4 8 11 12 2 6 6 6 5 2 6 6 6 5 2 4 8 11 12	1.5087e+03
104	820	1 6 5 4 3 4 3 1 3 4 3 1 1 6 5 4	None	5 7 8 9 5 6 6 6 5 6 6 6 5 7 8 9	39 24	5 7 8 12 5 7 8 11 1 1 1 1 5 7 8 11 5 7 8 12 2 4 8 12 12 2 6 6 6 5 2 6 6 6 5 2 4 8 12 12	1.5086e+03
105	838	1 6 5 3 3 4 3 0 3 4 3 0 1 6 5 3	8 12	5 7 8 9 5 6 6 7 5 6 6 7 5 7 8 9	8 16	5 7 8 12 5 7 8 12 1 1 1 1 5 7 8 12 5 7 8 12 2 4 8 12 12 2 6 6 6 5 2 6 6 6 5 2 4 8 12 12	2.6740e+03
106	840	1 6 5 3 3 4 2 7 3 4 2 7 1 6 5 3	None	5 7 8 9 5 6 6 7 5 6 6 7 5 7 8 9	29 34	5 7 8 12 5 7 8 12 1 1 1 1 5 7 8 12 5 7 8 12 2 4 8 12 12 2 6 6 7 5 2 6 6 7 5 2 4 8 12 12	1.5565e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
107	842	1 6 5 3 3 4 2 6 3 4 2 6 1 6 5 3	None	5 7 8 9 5 6 6 7 5 6 6 7 5 7 8 9	35 30	5 7 8 12 5 7 8 12 1 1 1 1 5 7 8 12 5 7 8 12 2 4 8 12 12 2 6 6 7 6 2 6 6 7 6 2 4 8 12 12	1.5093e+03
108	844	1 6 4 2 3 4 2 6 3 4 2 6 1 6 4 2	None	5 7 8 9 5 6 6 7 5 6 6 7 5 7 8 9	24 39	5 7 8 12 5 7 8 12 1 1 1 1 5 7 8 12 5 7 8 12 2 4 8 13 12 2 6 6 7 6 2 6 6 7 6 2 4 8 13 12	1.5080e+03
109	846	1 6 4 1 3 4 2 5 3 4 2 5 1 6 4 1	None	5 7 8 9 5 6 6 7 5 6 6 7 5 7 8 9	8 16	5 7 8 12 5 7 8 13 1 1 1 1 5 7 8 13 5 7 8 12 2 4 8 13 12 2 6 6 7 6 2 6 6 7 6 2 4 8 13 12	1.5068e+03
110	872	1 6 4 -1 3 4 2 5 3 4 2 5 1 6 4 -1	4 16	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	40 25 4 20	5 7 8 13 5 7 8 13 1 1 1 1 5 7 8 13 5 7 8 13 2 4 8 13 13 2 6 6 7 6 2 6 6 7 6 2 4 8 13 13	1.5463e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
111	874	1 6 4 9 3 4 2 5 3 4 2 5 1 6 4 9	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	4 20	5 7 8 14 5 7 8 13 1 1 1 1 5 7 8 13 5 7 8 14 2 4 8 13 13 2 6 6 7 6 2 6 6 7 6 2 4 8 13 13	1.5453e+03
112	876	1 6 4 8 3 4 2 5 3 4 2 5 1 6 4 8	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	40 25	5 7 8 14 5 7 8 13 1 1 1 1 5 7 8 13 5 7 8 14 2 4 8 13 14 2 6 6 7 6 2 6 6 7 6 2 4 8 13 14	1.5442e+03
113	878	1 6 3 7 3 4 2 5 3 4 2 5 1 6 3 7	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	39 24	5 7 8 14 5 7 8 13 1 1 1 1 5 7 8 13 5 7 8 14 2 4 8 14 14 2 6 6 7 6 2 6 6 7 6 2 4 8 14 14	1.5433e+03
114	880	1 6 2 6 3 4 2 5 3 4 2 5 1 6 2 6	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	39 24	5 7 8 14 5 7 8 13 1 1 1 1 5 7 8 13 5 7 8 14 2 4 8 15 14 2 6 6 7 6 2 6 6 7 6 2 4 8 15 14	1.5433e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
115	882	1 6 2 6 3 4 2 4 3 4 2 4 1 6 2 6	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	30 35	5 7 8 14 5 7 8 13 1 1 1 1 5 7 8 13 5 7 8 14 2 4 8 15 14 2 6 6 7 7 2 6 6 7 7 2 4 8 15 14	1.5422e+03
116	884	1 6 2 5 3 4 2 3 3 4 2 3 1 6 2 5	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	16 8	5 7 8 14 5 7 8 14 1 1 1 1 5 7 8 14 5 7 8 14 2 4 8 15 14 2 6 6 7 7 2 6 6 7 7 2 4 8 15 14	1.5414e+03
117	886	1 6 2 4 3 4 2 2 3 4 2 2 1 6 2 4	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	8 16	5 7 8 14 5 7 8 15 1 1 1 1 5 7 8 15 5 7 8 14 2 4 8 15 14 2 6 6 7 7 2 6 6 7 7 2 4 8 15 14	1.5414e+03
118	890	1 6 2 2 3 4 2 2 3 4 2 2 1 6 2 2	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	40 25 4 20	5 7 8 15 5 7 8 15 1 1 1 1 5 7 8 15 5 7 8 15 2 4 8 15 15 2 6 6 7 7 2 6 6 7 7 2 4 8 15 15	1.5400e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
119	892	1 6 2 1 3 4 2 2 3 4 2 2 1 6 2 1	None	5 7 8 10 5 6 6 7 5 6 6 7 5 7 8 10	4 20	5 7 8 16 5 7 8 15 1 1 1 1 5 7 8 15 5 7 8 16 2 4 8 15 15 2 6 6 7 7 2 6 6 7 7 2 4 8 15 15	1.5398e+03
120	918	1 6 2 0 3 4 2 2 3 4 2 2 1 6 2 0	4 16	5 7 8 11 5 6 6 7 5 6 6 7 5 7 8 11	25 40	5 7 8 16 5 7 8 15 1 1 1 1 5 7 8 15 5 7 8 16 2 4 8 15 16 2 6 6 7 7 2 6 6 7 7 2 4 8 15 16	1.5739e+03
121	920	1 6 1 11 3 4 2 2 3 4 2 2 1 6 1 11	None	5 7 8 11 5 6 6 7 5 6 6 7 5 7 8 11	24 39	5 7 8 16 5 7 8 15 1 1 1 1 5 7 8 15 5 7 8 16 2 4 8 16 16 2 6 6 7 7 2 6 6 7 7 2 4 8 16 16	1.5738e+03
122	926	1 6 1 8 3 4 2 1 3 4 2 1 1 6 1 8	None	5 7 8 11 5 6 6 7 5 6 6 7 5 7 8 11	40 25 20 16 8 4	5 7 8 17 5 7 8 16 1 1 1 1 5 7 8 16 5 7 8 17 2 4 8 16 17 2 6 6 7 7 2 6 6 7 7 2 4 8 16 17	1.5719e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
123	946	1 6 1 7 3 4 2 0 3 4 2 0 1 6 1 7	8 12	5 7 8 11 5 6 6 8 5 6 6 8 5 7 8 11	8 16	5 7 8 17 5 7 8 17 1 1 1 1 5 7 8 17 5 7 8 17 2 4 8 16 17 2 6 6 7 7 2 6 6 7 7 2 4 8 16 17	1.6011e+03
124	972	1 6 0 5 3 4 2 7 3 4 2 7 1 6 0 5	3 15	5 7 9 11 5 6 6 8 5 6 6 8 5 7 9 11	16 8 39 24 30 35	5 7 8 17 5 7 8 18 1 1 1 1 5 7 8 18 5 7 8 17 2 4 8 17 17 2 6 6 7 8 2 6 6 7 8 2 4 8 17 17	1.5808e+03
125	974	1 6 9 5 3 4 1 7 3 4 1 7 1 6 9 5	None	5 7 9 11 5 6 6 8 5 6 6 8 5 7 9 11	15 7	5 7 8 17 5 7 9 18 1 1 1 1 5 7 9 18 5 7 8 17 2 4 8 17 17 2 6 6 7 8 2 6 6 7 8 2 4 8 17 17	1.5752e+03
126	976	1 6 8 4 3 4 1 7 3 4 1 7 1 6 8 4	None	5 7 9 11 5 6 6 8 5 6 6 8 5 7 9 11	24 39	5 7 8 17 5 7 9 18 1 1 1 1 5 7 9 18 5 7 8 17 2 4 8 18 17 2 6 6 7 8 2 6 6 7 8 2 4 8 18 17	1.5752e+03

Continua na página seguinte

Tabela C.3 – continua da página anterior

Iteration	DOF	Beta (β)	Refined Loops	Loop's Order	Refined Edges	Edge's Order	Energy
127	978	1 6 8 3 3 4 1 6 3 4 1 6 1 6 8 3	None	5 7 9 11 5 6 6 8 5 6 6 8 5 7 9 11	16 8	5 7 8 17 5 7 9 19 1 1 1 1 5 7 9 19 5 7 8 17 2 4 8 18 17 2 6 6 7 8 2 6 6 7 8 2 4 8 18 17	1.5751e+03
128	980	1 6 8 2 3 4 1 6 3 4 1 6 1 6 8 2	None	5 7 9 11 5 6 6 8 5 6 6 8 5 7 9 11	40 25	5 7 8 17 5 7 9 19 1 1 1 1 5 7 9 19 5 7 8 17 2 4 8 18 18 2 6 6 7 8 2 6 6 7 8 2 4 8 18 18	1.5752e+03

